RESEARCH NOTE 80-13

LEVEL II

AD A092206

DESCRIPTION OF THE TACTICAL OPERATIONS SYSTEM INFORMATION FLOW MODEL

Gary Witus, Robert W. Blum, Mark Graulich

VECTOR RESEARCH, INCORPORATED
P.O. Box 1506
Ann Arbor, Michigan 48106

Human Factors Technical Area

U. S. Army
Research Institute for the Behavioral and Social Sciences

May 1980

Contract DAHC19-78-C-0027

80 11 25 041

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ARI Research Notes 80-13	2. GOVT ACCESSION NO.: AD-A092 206
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): Description of the Tactical Operations System Information Flow Model.	5. TYPE OF REPORT & PERIOD COVERED: Interim May 1979-May 1980
	6. PERFORMING ORG. REPORT NUMBER: VRI-ARI-3-FR79-3
7. AUTHOR(s): Gary Witus, Robert W. Blum, Mark Graulich	8. CONTRACT OR GRANT NUMBER(s): DAHC19-78-C-0027
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Vector Research, Incorporated, PO Box 1506, Ann Arbor, Michigan 48106	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 2Q263739A793, Task 1A Work Unit 001
11. CONTROLLING OFFICE NAME AND ADDRESS: CORADCOM Field Office, ATTN: DRDCO-FL, Ft. Leavenworth, Kansas 66027	12. REPORT DATE: 30 November 1979
	13. NUMBER OF PAGES: 83
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): US Army Research Institute for the Behavioral and Social Sciences, ATTN: PERI-OS, 5001 Eisenhower Ave Alexandria VA 22333	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for open release; distribution unlinited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was technically monitored by Mr. Robert S. Andrews and Dr. Stanley M. Halpin of ARI and LTC Larry Walker of CORADCOM.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Command and Control	Information Management	Standing Operating Procedures (SOP)
Communications	Information Summarization	Tactical Operations System (TOS)
Computers	Mathmatical Modeling	
Data base Management	Networks	
Design aid	Queueing	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes a mathematical model of the Division Tactical Operations System (DIV TOS) ~~developed during the second~~ phase of a project to develop information management concepts and procedures for automated battlefield command and control systems. The research effort evolved from previous work to develop a design/decision aid (DDA) for the evaluation of alternative information management polices. The original DDA model was concerned only with the Division Computing Center. The

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Abstract (continued)

original effort was expanded to encompass both the distributed processors -- the Tactical Computer Systems and Tactical Computer Terminals--- and the supporting communications. The resulting model not only provides a tool for the analysis of TOS and its component parts, but also has the potential for application to other distributed command support systems with a central node and data base.

PREFACE

This document is one of eight reports which describe the work performed by Vector Research, Incorporated (VRI) and its subcontractor, Perceptronics, Incorporated, for the US Army Research Institute for the Behavioral and Social Sciences (ARI) under the second phase of contract number DAHC19-78-C-0027. The work described was performed over 12 months of an anticipated 36-month three-phased project. The overall objective of the project has been to produce procedural guidelines to be used by divisions in the field in developing standard operating procedures for information management in the Tactical Operations System (TOS). As a consequence of the redirection of the TOS development effort in November 1979, the objective of this work was reinterpreted to include automated battlefield command control systems (ABCCS) in general, using TOS for an explicit example of the design, human factors, and management control considerations which must be addressed.

The VRI study team for phase II was comprised of Dr. Robert W. Blum (Project Leader), Ms. Cathleen A. Callahan, Dr. W. Peter Cherry, Mr. Mark G. Graulich, Mr. Donald Kleist, Mr. Mark Meerschaert, Mr. Gregory Touma, and Mr. Gary Witus. The Perceptronics team for phase II consisted of Dr. Michael G. Samet and Dr. Ralph E. Geiselman.

The authors wish to acknowledge the helpful contributions of Dr. Stanley M. Halpin and Mr. Robert Andrews, who were charged with monitoring the study for ARI; and LTC L. Walker, MAJ. A. Edmonds, and Mr. M. Carrio, who performed a similar function for that portion of the study effort which was jointly sponsored with ARI by the US Army Communications Research and Development Command (CORADCOM).

The eight reports are as follows:

Blum et al. **Information Management for an Automated Battlefield Command and Control System: Executive Summary.** ARI Research Report 1249. -- presents an overview of the project and the other seven reports.

Callahan et al. **Guidelines for Managing the Flow of Information in an Automated Battlefield Command and Control System.** ARI Research Report 1348. -- describes considerations in and procedures for the management of contemporary ABCC systems.

Geiselman and Samet. **Guideline Development for Summarization of Tactical Data.** ARI Technical Report 458. -- an analysis of procedures for the extraction, summarization, and presentation of critical information.

Witus et al. **Analysis of Information Flow in the Tactical Operations System (TOS).** ARI Research Notes 80-12. -- describes the purpose, approach, and results of a TOS analysis which focused on TOS when integrated with a planned communications support system.

Witus et al. **Description of the Tactical Operations System Information Flow Model.** ARI Research Notes 80-13. -- describes the representation of TOS used to develop the analysis package and the mathematics of the model.

Witus et al. **User's Manual for the Tactical Operations System Analysis Package.** ARI Research Notes 80-14. -- explains the use and operation of the analysis package.

Witus et al. **Programmer's Manual for the Tactical Operations System Analysis Package.** ARI Research Notes 80-15. -- describes the programming details of the package to facilitate modifications or transfer between host systems.

Cherry, W. **All Source Analysis System: Design Issues.** ARI Working Paper HF80-XX. -- a discussion of design issues associated with the emerging ASAS concept.

CONTENTS

EXHIBITS

EXHIBITS

(Concluded)

1.0 INTRODUCTION

The purpose of this report is to document a mathematical model of the Division Tactical Operations System (DIV TOS). The research effort evolved from previous work to develop a design/decision aid (DDA) for the evaluation of alternative information management policies.[1] The original DDA model was concerned only with the Division Computing Center. The original effort was expanded to encompass both the distributed processors -- the Tactical Computer Systems and Tactical Computer Terminals -- and the supporting communications. The resulting model not only provides a tool for the analysis of TOS and its component parts, but also has the potential for application to other distributed command support systems with a central node and data base.

The report is organized into three chapters and supplemented by an appendix. The three chapters are: (1) Introduction; (2) TOS Representation; and (3) Equations and Data Requirements. This organization corresponds to the historical development of the model. Chapter 1.0, Introduction, presents a brief background description of TOS and describes the purpose and scope of the model. Chapter 2.0, TOS Representation, describes TOS in terms of its functions and subsystems. The model equations are based on this representation of the system. Chapter 3.0, Equations and Data Requirements, presents the model equations and their inputs. Appendix A, Queueing Systems with Pooled Arrival Streams, discusses the adequacy of certain approximations made in the representation of queueing phenomena which would occur during TOS operation. Appendix B is a glossary of acronyms.

---

[1] See Information Management for the Tactical Operations System (TOS), ARI Research Report 1228, October 1979.

1.1 BACKGROUND

The A-Specs provide a general description of TOS:

"The TOS is intended to be a secure, automatic data processing system serving the command and staff elements of the Division at the Tactical Operations Center (TOC), Tactical Command Post (TAC CP), subordinate Brigade Command Post (BDE CP), subordinate Battalion Command Post (BN CP), a subordinate Armored Cavalry Squadron and support liaison points. The system would provide the capability to aid the commanders in controlling and processing, storing, retrieving and disseminating information concerning the status and location of friendly and enemy units. The TOS would be secure, modular, and would provide for commonality and interchangeability of hardware components among its functional areas and with other Army tactical systems. In non-tactical deployment, the system would have the capability to permit training of user personnel without affecting its mission-ready capability.

"The primary mission of TOS would be to provide the commander and his staff, in a timely manner, the operations and intelligence information that they require to: see the battlefield; make decisions to exploit enemy force weaknesses; and, determine courses of action for the effective employment of friendly resources. As a command and control system, TOS would have a secondary mission to function as the focal point for the exchange of data with other tactical data systems.

"The TOS would operate in a mid to high intensity Warfare environment. The critical formidable threat is expected to be highly mobile, numerically superior, armored and mechanized forces. The critical technical threat to TOS is expected to be electronic warfare (EW) operations oriented toward analyzing the system and its

communications with the intent of determining information content or degrading the TOS communications or the operation of the system itself. TOS would counter the critical formidable threat by providing the Division Commander and his staff near real-time information concerning the tactical situation. TOS would counter the critical technical threat by techniques which nullify or resist the threat."[1]

1.2 PURPOSE

The purpose of the TOS model is to provide a means to predict the steady state system performance from a set of inputs describing: (1) the engineering characteristics; (2) the network configuration; (3) the operation procedures; (4) the user demand; and (5) the environmental factors. The steady state system performance is the performance level to which the system will converge over time when operating under stable conditions. Stable conditions imply no ad hoc changes in the network configuration, the user demand, or other model inputs. The system performance measures are the utilization, expected queue length, and expected delay at selected system components.

- Utilization is the fraction of time a component is busy.
- Expected queue length is the average number of messages waiting for service at a component.
- Expected delay at a component is the sum of the average service time for a message and the average time a message spends awaiting service.

---

[1] System Specifications for the Division Tactical Operations System (TOS), CO-SS-3000-TO, April 1979.

1.3 SCOPE

Due to the limitations in the time and data available, it was necessary to restrict the scope of the model. Several restrictions were placed on the scope of the model. First and foremost, the model would represent TOS as a system of interacting components, but would not incorporate engineering models of the components. Instead, the engineering characteristics of the components would be inputs to the system model. Second, the performance measures would be computed only for the major system components: (1) the communications nets; (2) the Data Base Processor (DBP) Central Processing Unit (CPU); (3) the Front End Processor CPU; (4) the CPU of each Tactical Computer System (TCS); (5) the CPU of each Tactical Computer Terminal (TCT); and (6) the two disk controllers in the Division Computing Center (DCC). Third, within a given network configuration each user was assumed to have a single route connecting him with the DCC. Fourth, all messages were assumed to be of equal priority. Fifth, the model would not represent finite buffers at processors. Sixth, human factors were not incorporated into the model. (Consequently, the model computes only equipment delays and does not address the issue of whether or not humans could or would use the system efficiently.) Seventh, the model was based on the A-specs,[1] and draft B-specs.[2] These sources provided an incomplete description of TOS. As a consequence, in order to complete the model, various specific assumptions had to be made regarding the design and operation of TOS.

[1] System Specifications for the Division Tactical Operations System (DTOS), CO-SS-3000-TO, April 1979.

[2] Computer Program Configuration Item Specification Network Communication Processing for Division Tactical Operations System (DTOS), CR-CS-0002-B12, May 1979.

Also, approximation formuli were used in the model when it was known that the data required by more accurate formuli would not be available. These assumptions and approximations are presented as a part of the discussions of the TOS representation and model equations in chapters 2.0 and 3.0.

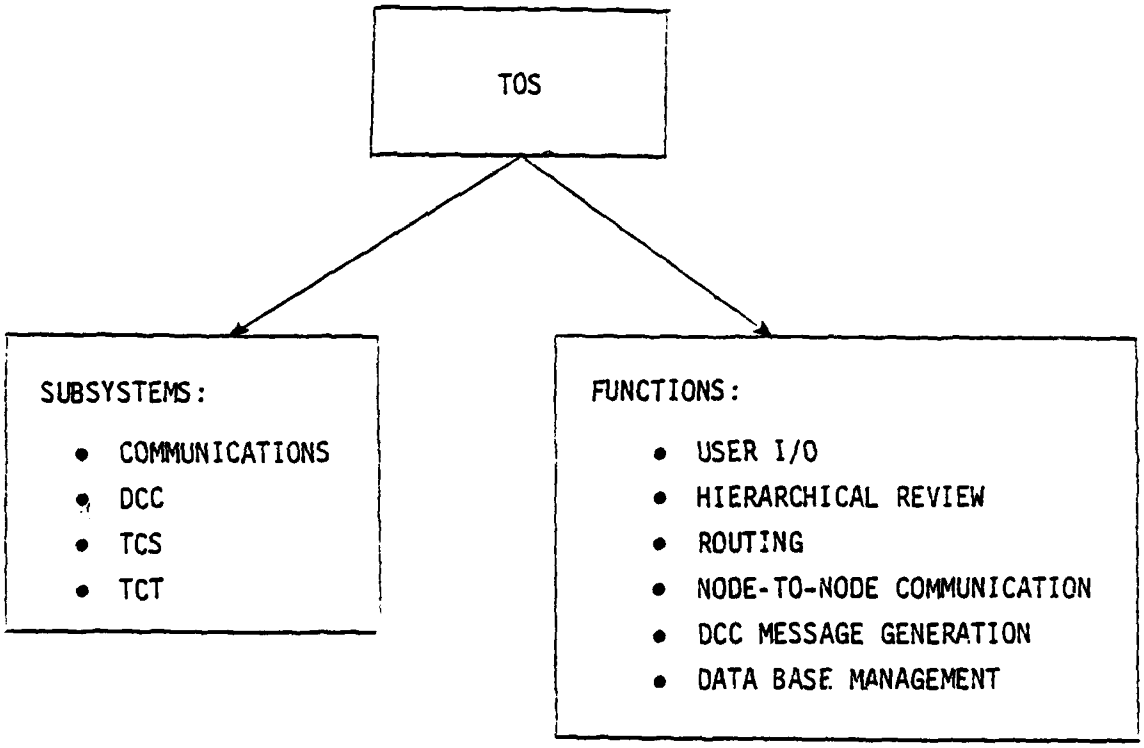
2.0 TOS REPRESENTATION

TOS consists of subsystems which interact to perform a set of functions. The decomposition of TOS into functions and subsystems which was used to develop the model equations is presented in exhibit 2-1. The first section of this chapter discusses these functions and their representation in the model. The second section discusses the subsystems and their operations in carrying out the system functions.

2.1 FUNCTIONS

There are six primary functions: (1) user I/O; (2) hierachical review; (3) message routing; (4) node-to-node communication; (5) message generation at the DCC; and (6) data base management. User I/O refers to all interactions between the user and the local terminal including output message formating and display, prompting, reading input, and error checking. Hierarchical review is a means to control the quality of information entering the data base. It involves automatic checking of data base updates from battalions against predefined criteria at brigade and human review if the criteria are met. Message routing is the process of establishing the sequence of nodes through which the message will travel enroute to its final destination -- either the DCC or a user. Routing is based on the network configuration. In accordance with the scope of the model, only a single route (comprised of any number of nodes in sequence) between each user and the DCC is represented. Node-to-node communication is the process of transferring a correct copy of a message from one node to another and acknowledging the successful transfer. Node-to-node communication includes message transmission, retransmissions, transmission of ACKs and NAKs, communication security,

EXHIBIT 2-1: TOS DECOMPOSITION

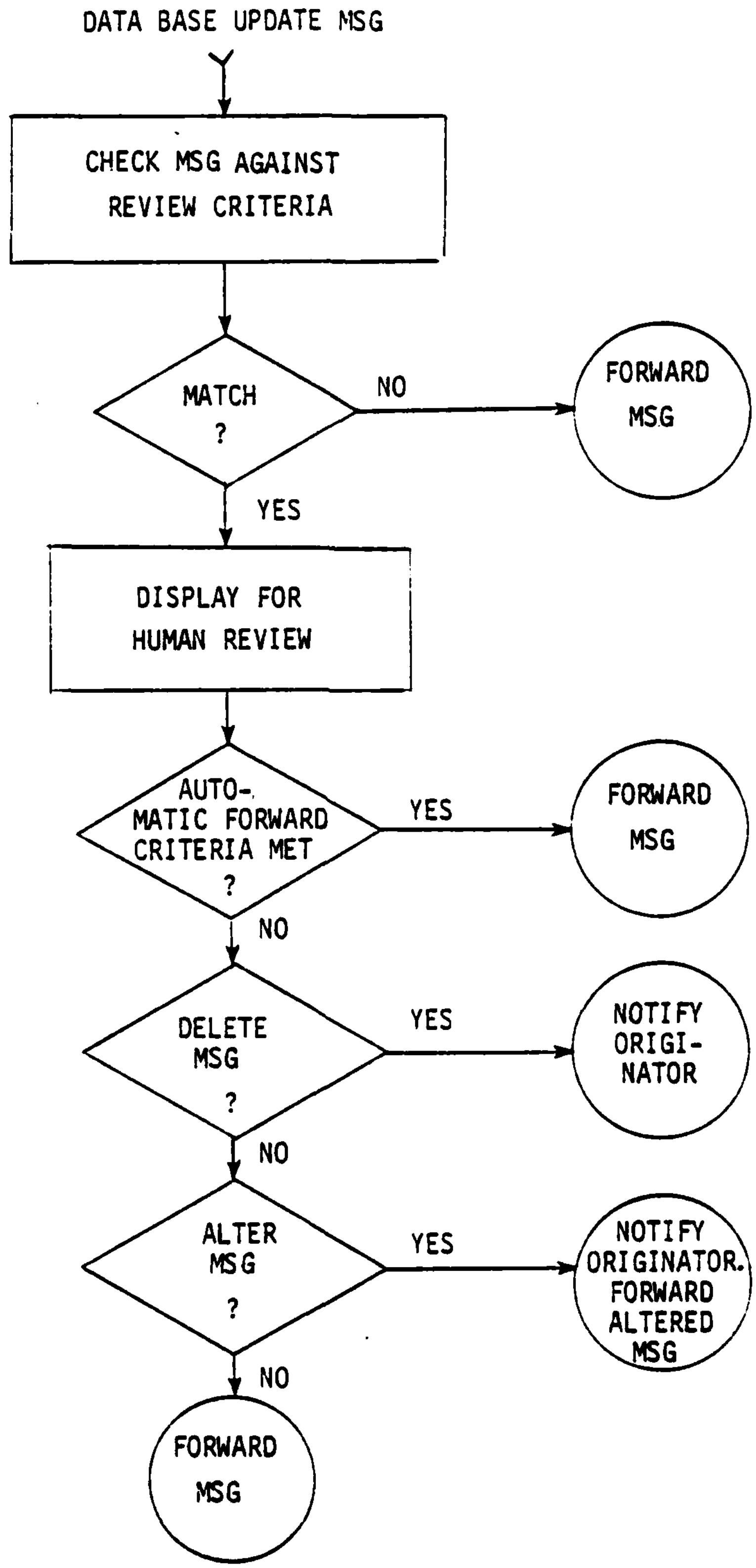


and error detection and correction (EDC) procedures. Message generation at the DCC refers to all processes which produce output messages (excluding link control/response messages such as ACKs and NAKs). These include ad hoc queries, filters, correlations, thresholds, preloaded queries, and standing requests for information (SRI). Data base management refers to maintaining and updating the data base at the main DCC. This involves updating the inverted key file both in core and on disk and updating the file proper on disk. Three functions: (1) hierarchical review and filtering; (2) node-to-node communication; and (3) DCC message generation are described in greater detail below.

2.1.1 HIERACHICAL REVIEW

A schematic representation of the review function is presented in exhibit 2-2. The process begins when a message which would update a file in the data base arrives at the brigade enroute to the DCC. The brigade reviewer will have already defined criteria which, if matched by an incoming message, will cause the message to be displayed to him for review. If the message does not match the review criteria, it is forwarded to the next stage of its processing. If the message is posted for review, it can be automatically forwarded if either of two criteria are met: (1) the message waits for some predefined amount of time without being reviewed (the time threshold being less than ten minutes); or (2) ten additional messages waiting for review pile up behind the first message. If the message is displayed for review, one of three things can happen to the message: (1) it can be forwarded unchanged; (2) it can be altered and forwarded; or (3) it can be deleted. If the message is altered, a copy of the altered version is sent to the message

EXHIBIT 2-2: SCHEMATIC REPRESENTATION OF HIERARCHICAL REVIEW



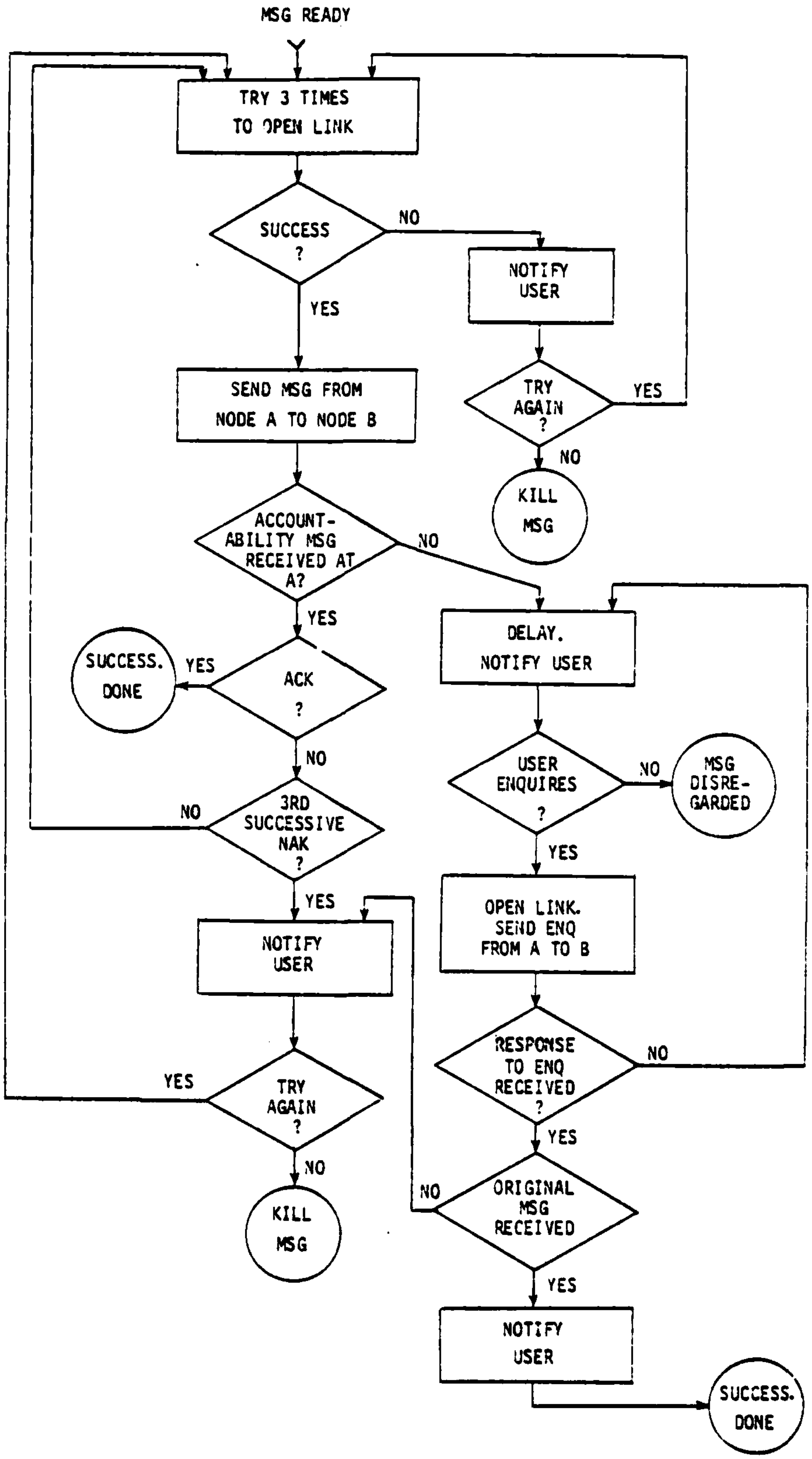
originator as well as to the DCC. If the message is deleted, the originator is notified and is sent a copy of the deleted message. As represented in the model, for an update to file-j being reviewed at node-i, there is a probability P_{1ij} that the update is deleted, and a probability P_{2ij} that it is altered. Setting either or both of these probabilities to zero causes the model to preclude the associated action from happening.

2.1.2 NODE-TO-NODE COMMUNICATION

A schematic overview of the node-to-node communication function is presented in exhibit 2-3. Node-to-node communication begins when the message is ready for transmission, i.e., Hamming Code has been applied to the message, time dispersed coding (TDC) has been applied, other EDC procedures, if used, are applied, and the various pilots, headers, and trailers have been attached to the message. The transmitting node then attempts to open a line to the next node on the route.[1] If the line is occupied, the device waits a specified time, then tries again. The longer the node waits between tries, the lower the probability of subsequently detecting the line to be occupied. If the line is detected to be busy three times in succession, the user is notified of the condition, and the terminal waits for instructions. If he says to try again, the process is repeated. Otherwise, the message is killed. Once the link is opened, the message, with its headers and trailers, is transmitted. If the originating node does not receive an accountability

[1]Due to insufficient available documentation, the model does not represent time slotting on shared FM nets. Instead, free competition is assumed. Multichannel and cable users have dedicated lines.

EXHIBIT 2-3: SCHEMATIC OVERVIEW OF NODE-TO-NODE COMMUNICATION

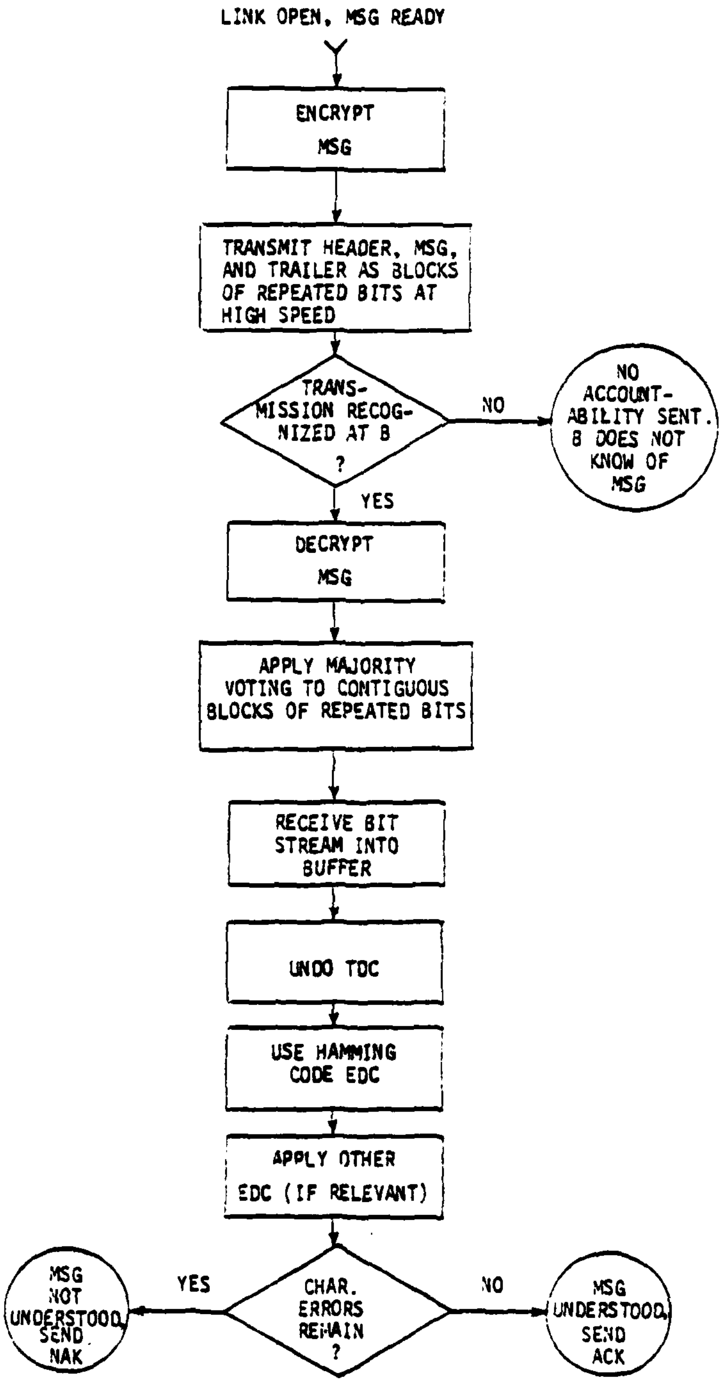


message (ACK/NAK) within some delay time, the user is notified. If he is interested, he sends an enquire (ENQ) message. If the ENQ generates a response, the response will inform the originator that the message was ACKed or NAKed. If an accountability message is received and it is an ACK, the function has been successfully executed. If an accountability message is received and it is a NAK but not the third successive NAK, the node automatically attempts to send the message again. If three successive NAKs have been received, or a NAK after an ENQ, the user is notified and the terminal waits for instructions. If he says to try again, the process starts over. Otherwise, the message is killed.

Exhibit 2-4 shows a detailed expansion of the process of sending a message from one node to another. Once the message is ready and the link is open, sending a message from one node to another begins with encryption. As the message is encrypted, the encrypted portion is transmitted. Each bit of the header, message body, and trailer is transmitted at high speed as a contiguous block of identical bits. For example, on FM each data bit is transmitted as 13 identical bits on a 16 kbps carrier. When the message is received, majority voting will be applied to the block of 13 bits. Consequently, the net transmission rate is 1.2 kbps.

It may be that the message will not be recognized at the receiver. Two events could cause this to occur. First, if a sufficiently high error rate is experienced on the high speed bits of the keying sequence, the receiving equipment may fail to recognize that a message is being transmitted. Second, if the time required for signal recognition, power-up, and phasing and synchronization at the receiving node (and all retransmission stations between the transmitting and receiving nodes)

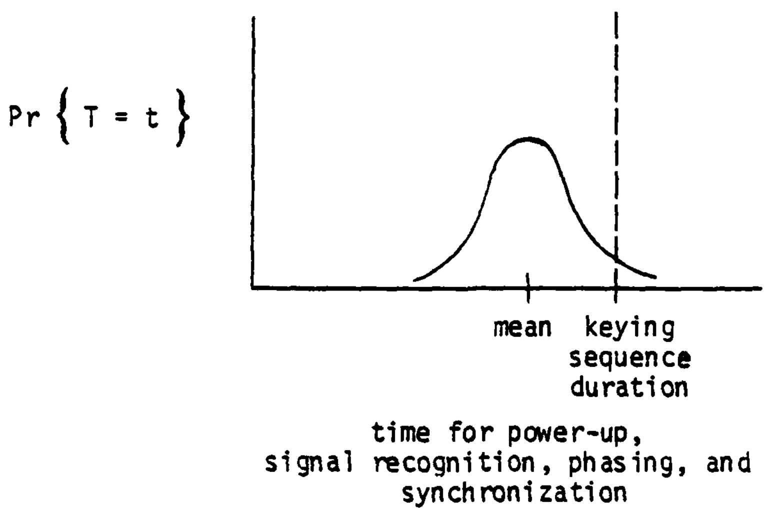
EXHIBIT 2-4: SCHEMATIC REPRESENTATION OF SEND MESSAGE
(DETAIL OF NODE-TO-NODE COMMUNICATION)



exceeds the duration of the keying sequence, then the receiver will not recognize that a message has been transmitted.[1] Should this occur, no accountability message will be sent to the originating node. If the receiving node does recognize the message, then it is decrypted. If a hardware integrator is being used for the majority voting, this takes place immediately following decryption and prior to putting the message in a buffer. If software is being used for majority voting, then the message is first put into a buffer. The purpose of the majority voting is to reduce the apparent error rate on the message bits. If the signal experiences true white noise, then an error rate of one in ten on the high speed bits will produce an error rate of approximately one in ten

---

[1]The time required for these events is a random variable with some distribution. Consequently, if the keying sequence duration is only slightly greater than the mean, there will be a significant probability of non-recognition:



The mean and standard deviation of the distribution depend on the communications equipment involved, especially the number of retransmission stations for an FM net.

thousand after majority voting.[1] Unfortunately, most noise has considerable serial correlation, i.e., it does not resemble white noise. Majority voting is most effective in countering random noise. Consequently, its efficiency in dealing with serially correlated noise is a function of the particular environment and cannot be established in general. Time distributed coding, TDC, is used to control serially correlated noise. After majority voting has been applied, the TDC process is inverted to restore the message bits to their original order so that the characters are now contiguous blocks of 12 bits: seven information and five parity bits. TDC does not reduce the bit error rate, but causes the distribution of bit errors within a character to resemble that of a white noise process, i.e., serially uncorrelated. In the model, it is assumed that the distribution of bit errors after TDC is inverted is the same as that which would be caused by white noise: the probabilities that bits are in error are independent, identical, and independent of whether the bit was a 0 or a 1. After the TDC is inverted, the Hamming Code error detection and correction (EDC) is applied to each character. The effects of this EDC procedure are: (1) to produce a correct seven bit character if not more than one bit out of the 12 is in error; (2) to detect, but not correct, the error condition if more than two bits are in error. (When Hamming Code is used, a character consists of 12 bits, so a 1.2 kbps net transmits 100 characters.

---

[1] If there is true white noise, the error rate after majority voting on 13 bits equals

$$\left(1 - \sum_{i=0}^{6} \binom{13}{i} e^{i} (1-e)^{13-i}\right)$$

per second.) Next, any other EDC procedures such as multiple blocking[1] or an automatic retransmission on request (ARQ) scheme[2] would be applied. If, after the application of any other EDC procedures, there are still characters in error, the message will be NAKed. Otherwise, it will be ACKed, signifying successful communication.

Sending link control/response messages such as an ACK, NAK, or ENQ follows much the same pattern as other messages. The only difference, aside from message content, is that these messages do not generate accountability responses. That is, an ACK will not generate another ACK. This avoids the situation where ACKs generate ACKs which generate ACKs, and so on ad infinitum.

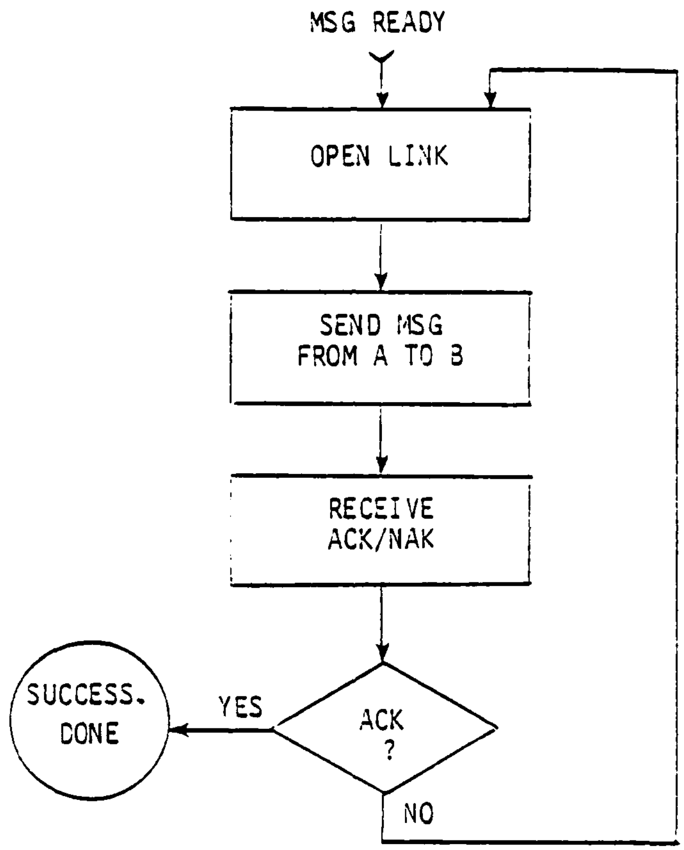
In that version of the model which was implemented for performing the analyses referenced earlier, it is assumed that: (1) the terminal waits only an infintesimally small time between each of its three successive attempts to get a line; (2) the user always instructs the machine to try again both to get a line and to transmit; and (3) the originating node always receives acountability messages, i.e., transmissions are not lost. These assumptions lead to the simplified schematic diagram of node-to-node communication shown in exhibit 2-5.

---

[1]Multiple blocking is an EDC procedure which consists of sending several copies of the same message under one header. Each copy goes through the majority voting, TDC, and Hamming Code sequence. A composite message is then made out of the several copies. A character in the composite message will be correct if that character is correct in any of the several copies. A character in the composite message will be in error only if that character is in error in all copies of the message.

[2]ARQ schemes constitute a class of EDC procedures. One ARQ scheme was represented in the model. In this approach, the receiving node will save a copy of the message even if characters are in error. The message is NAKed, and when the next copy is received it is compared to the retained copy to see if any characters which were in error on the retained copy are correct on the retransmitted copy. Characters that can be corrected are corrected. In this way, the best composite message is formed and retained. If the best composite message still has characters in error, another NAK is sent and the process is repeated.

EXHIBIT 2-5: SIMPLIFIED VIEW OF NODE-TO-NODE COMMUNICATION



2.1.3 DCC MESSAGE GENERATION

Message generation at the DCC refers to output messages, not link control/response messages. This function is schematically represented in exhibit 2-6. The process is straightforward and has a simple probability representation in the model. For each user-i and file-j there are two inputs: E_{1ij}, E_{2ij}. E_{1ij} is the expected number of outputs to user-i that occur as a result of an update to file-j from users other than user-i. These outputs are produced by SRI, filters, thresholds, and correlations. E_{2ij} is the expected number of outputs to user-i that occur as a result of a query by user-i against file-j.

2.2 SUBSYSTEMS

TOS has four major subsystems: (1) communications; (2) DCC; (3) TCS; and (4) TCT. These subsystems are identified within the Division TOS network configuration in exhibit 2-7. Each subsystem has one or more tasks which it performs during normal system operation. The tasks performed by the subsystems are shown in exhibit 2-8. For example, the TCT tasks are to get messages into and out of the system. This section discusses the hardware structure of each subsystem and the flow of activities through which the subsystems perform their tasks.

2.2.1 COMMUNICATIONS SUBSYSTEM

The communications subsystem includes all communications links between network nodes. These links may be cable, multichannel, or FM. The communications subsystem interfaces with other subsystems through the CIM or I/O port of the DCC, TCS, or TCT subsystem. Exhibits 2-9, 2-10, and 2-11 show the hardware structures of cable, multichannel, and FM

EXHIBIT 2-6: DCC MESSAGE GENERATION ARRIVES AT DBP

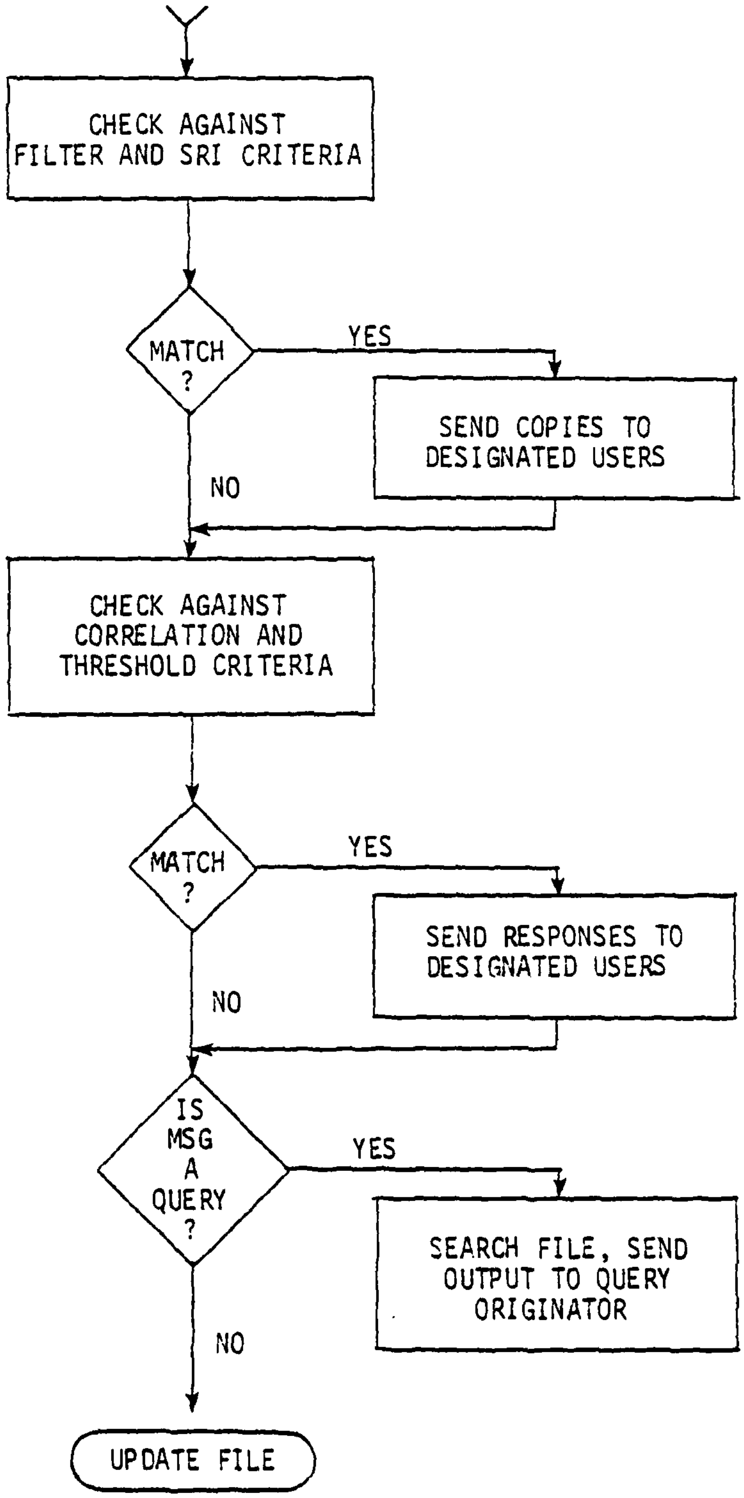
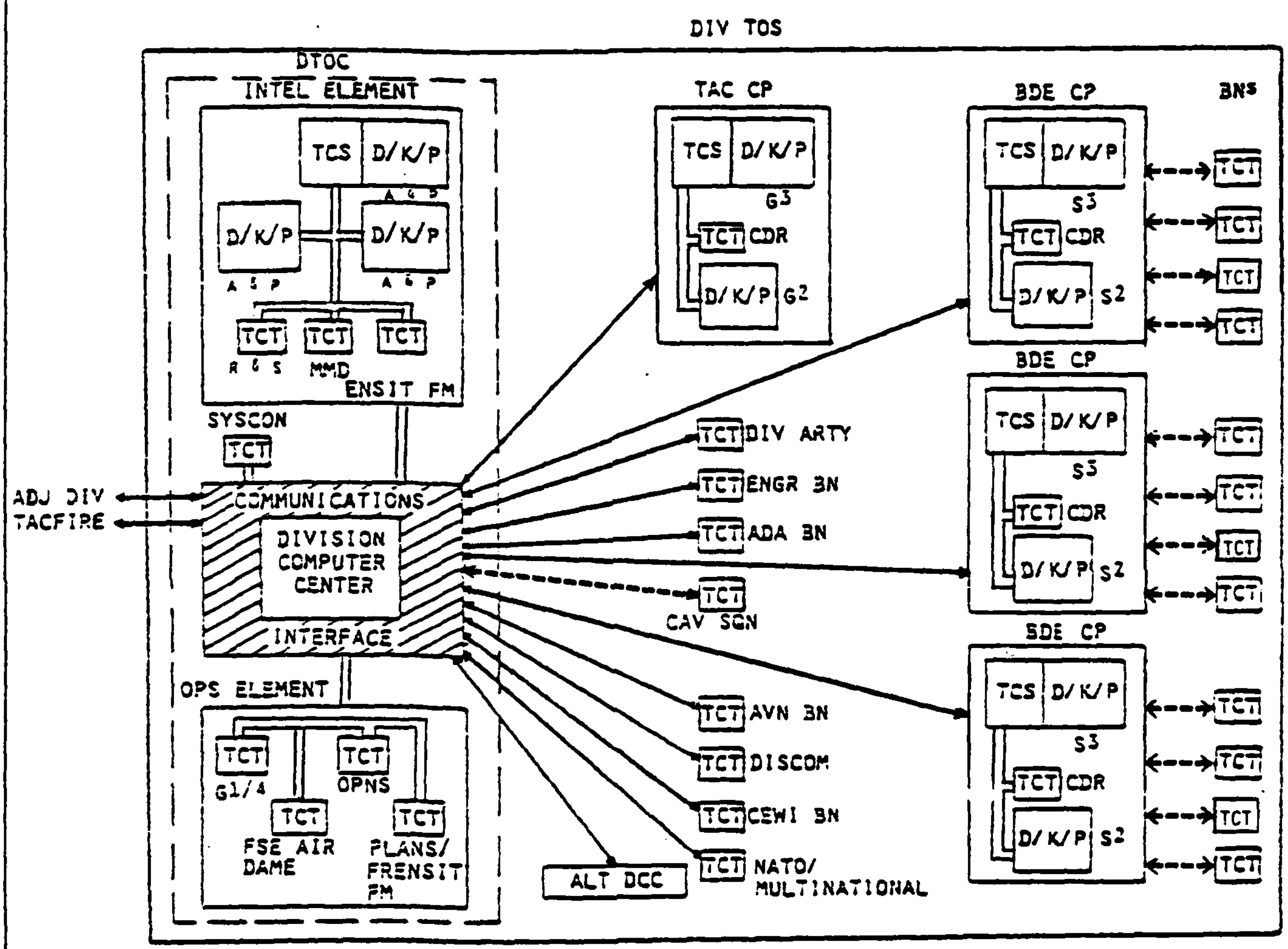
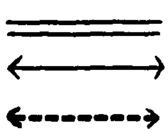


EXHIBIT 2-7: DIVISION TOS NETWORK CONFIGURATION



LEGEND:



CABLE (FDX)

MULTICHANNEL (FDX)

FM (HDX)

Source: A-Specs, April 1979

EXHIBIT 2-8: SUBSYSTEM TASKS

SUBSYSTEM	TASKS
COMMUNICATIONS	• COMMUNICATING BETWEEN NODES
DCC	• PROCESS ARRIVING MESSAGES
TCS	• ORIGINATING MESSAGES • TERMINATING MESSAGES • TRANSFERRING A MESSAGE TO DCC • TRANSFERRING A MESSAGE FROM DCC
TCT	• ORIGINATING MESSAGES • TERMINATING MESSAGES

EXHIBIT 2-9: CABLE EQUIPMENT CONNECTIONS

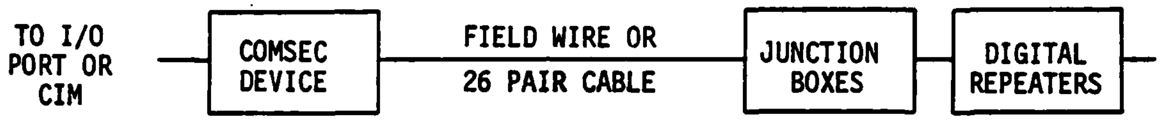
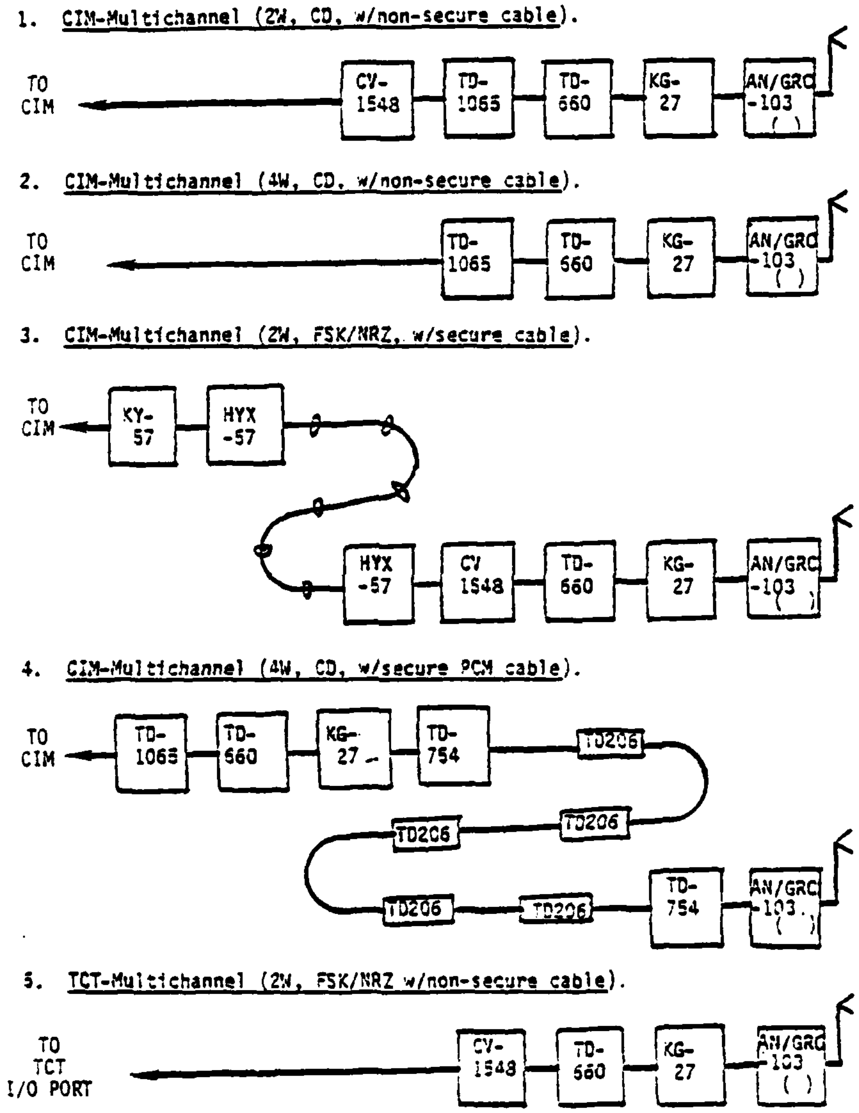
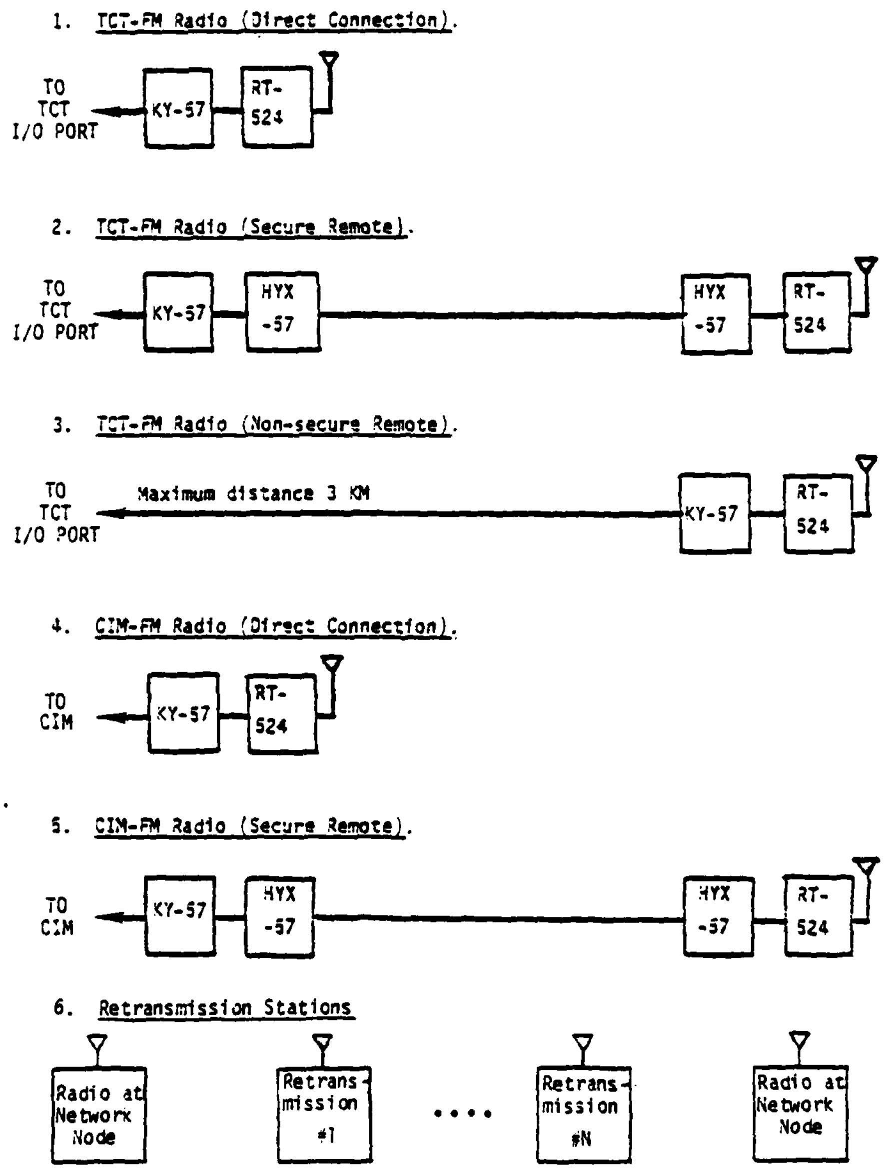


EXHIBIT 2-10: MULTICHANNEL EQUIPMENT CONNECTIONS



Source: Division Tactical Operations System Communications Plan (Annex G), May 1979

EXHIBIT 2-11: FM RADIO EQUIPMENT CONNECTIONS



Source: Division Tactical Operations System Communications Plan (Annex G), May 1979

links, respectively. The components of the subsystem for which performance measures are computed are the node-to-node cable, FM, and multichannel links.

The communications subsystem performs a single task: communicating between nodes. There is also a single activity performed: transmission of a message between nodes. A flow diagram of the operation of a communication link in performing its task is shown in exhibit 2-12.

2.2.2 DCC

The DCC subsystem includes the hardware and software at the main DCC. The hardware structure is presented in exhibit 2-13. The components for which the performance measures are computed are shown in shaded boxes in the exhibit.

The DCC performs six activities: (1) message intake; (2) internal accountability maintenance; (3) message checking; (4) updating the data base; (5) searching the data base; and (6) producing output messages. Intake refers to getting a message into the DCC; this is handled by the FEP. Maintaining internal accountability refers to making sure a message is not lost, or its stage in processing forgotten in a multitasking multiprocessor environment. To accomplish this accountability, the message and record of its status is maintained on the disks. Checking messages refers to matching incoming messages against templates to see if the message satisfies any SRI, filter, threshold, or correlation criteria. Updating the data base refers to updating the inverted key file and record file on the data disk and in core. Searching the data base refers to searching the inverted key file and, possibly, the record file. These three activities are handled by the DBP and the data disk

EXHIBIT 2-12: OPERATION OF A COMMUNICATIONS LINK

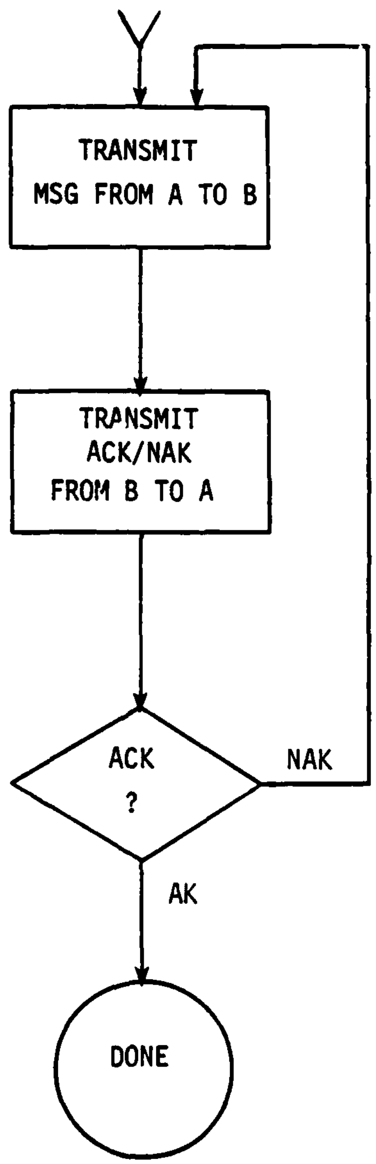
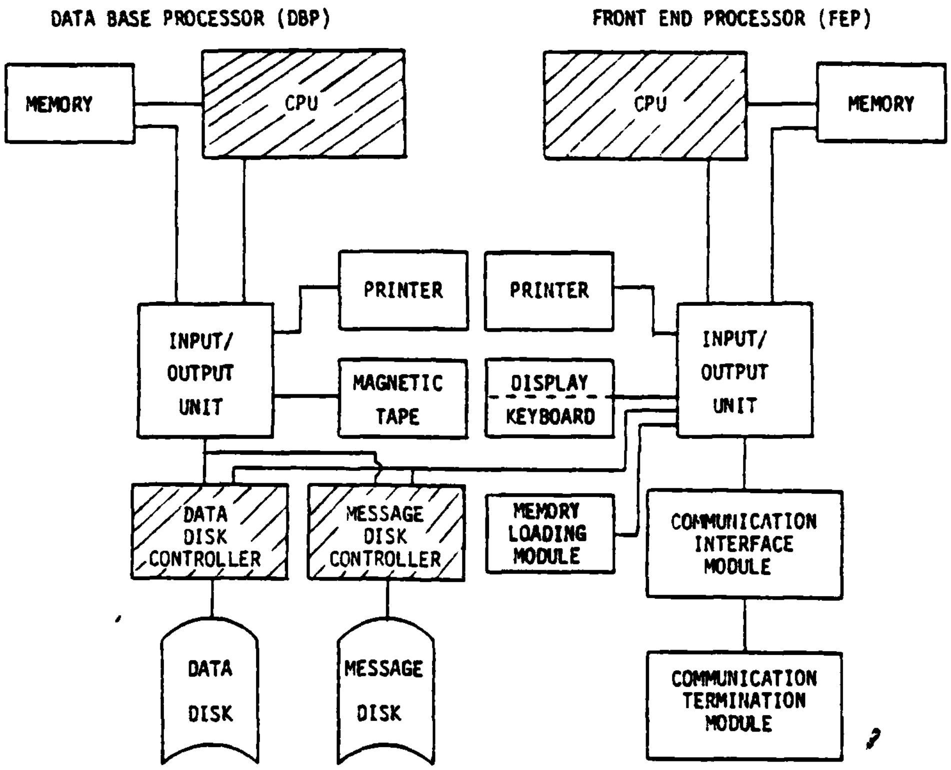


EXHIBIT 2-13: DIVISION COMPUTER CENTER (DCC) SUBSYSTEM



Source: <u>A-Specs, April 1979</u>

controller. Producing messages refers to compiling output messages and posting them for dissemination. This is performed by the FEP. The flow chart in exhibit 2-14 shows how these activities combine to perform the DCC task of processing arriving messages.

2.2.3 TCS

The TCS subsystem includes all TCS nodes in the network. The hardware structure of a TCS node is shown in exhibit 2-15. The component for which the performance measures are computed is in a shaded box.

The TCS performs four tasks: (1) originating a message; (2) terminating a message; (3) transferring a message to the DCC; and (4) transferring a message from the DCC. Originating a message refers to interfacing with the user and communicating the message to the next node. Terminating refers to receiving a communication as the designated receiving node, formatting it and displaying it. Transferring to the DCC refers to receiving a message from a subordinate node, reviewing it, and, possibly sending it on to the DCC. Transferring from the DCC refers to receiving a message designated for a subordinate node and sending it on its way.

The TCS subsystem performs these tasks through a combination of five activities: (1) inputting; (2) sending; (3) receiving; (4) reviewing; and (5) outputting. Inputting refers to the processes which get a message into the system and ready for communication to the next node. Sending refers to all processes which are repeated each time a message is transmitted or retransmitted from the node. Receiving refers to all processes which are repeated each time a message is transmitted or retransmitted to the node. Reviewing refers to checking messages against the hierarchical review criteria. Outputting refers to the processes by

EXHIBIT 2-14: FLOW CHART OF DCC PROCESSING OF ARRIVING MESSAGES

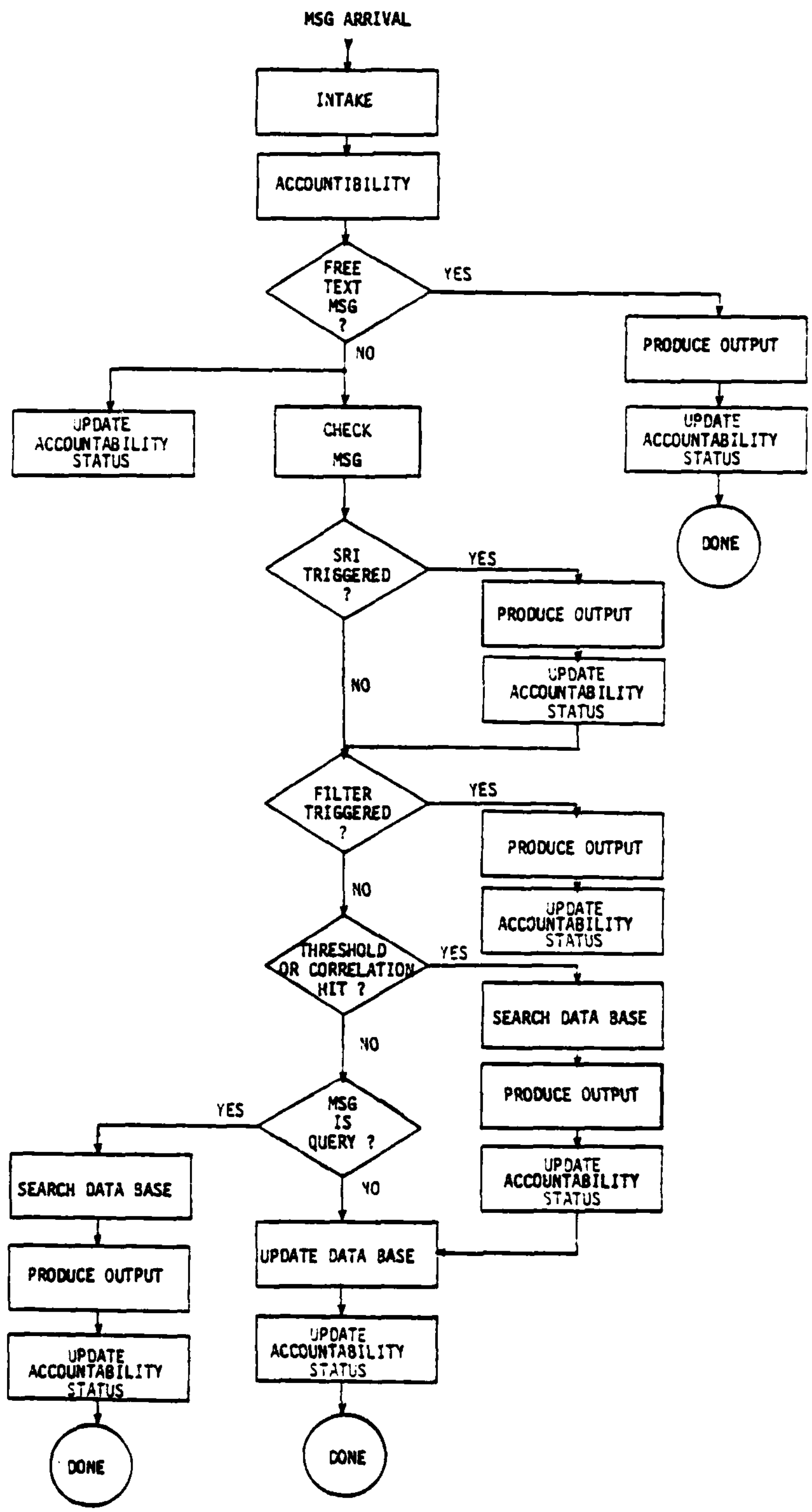
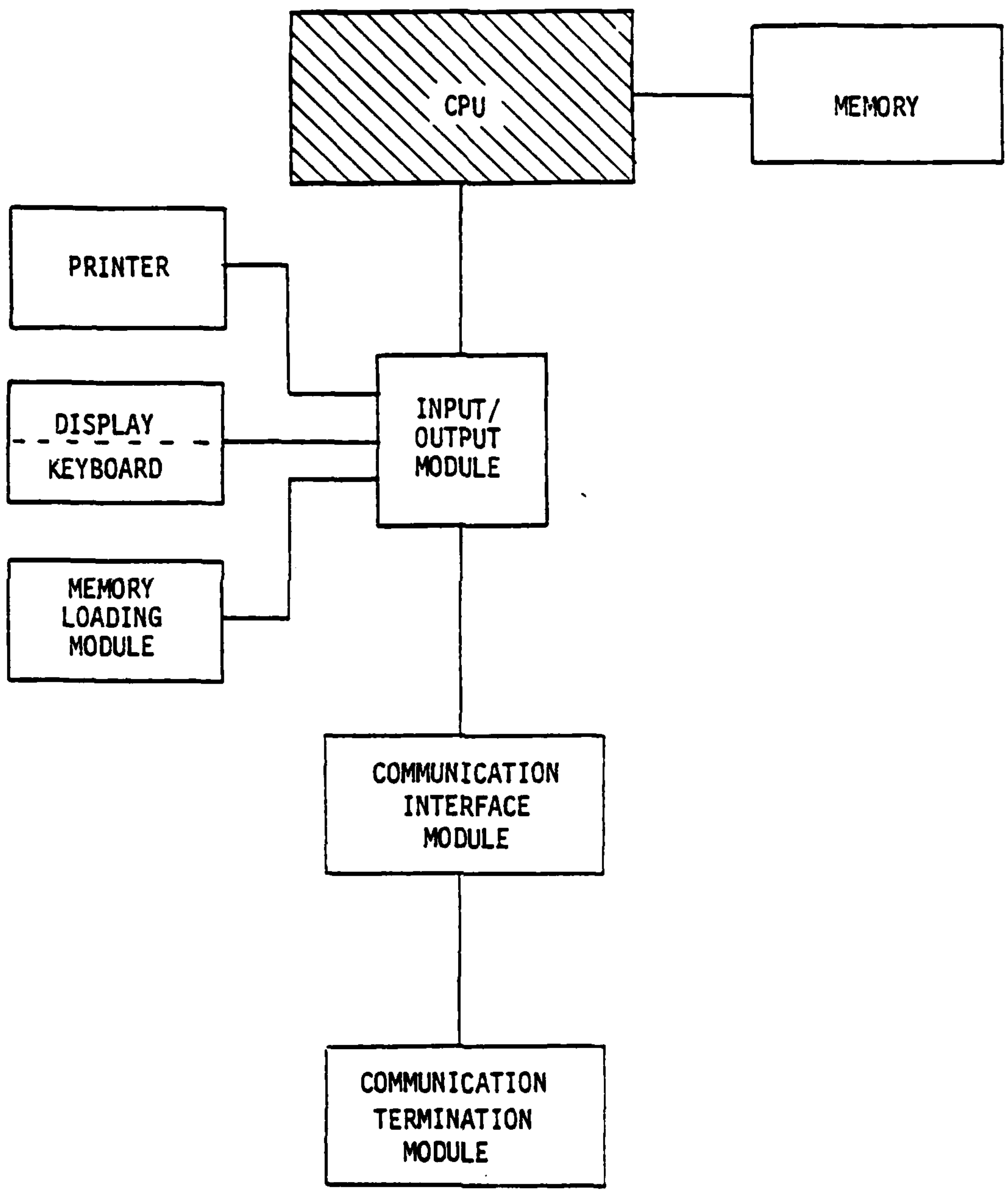


EXHIBIT 2-15: TACTICAL COMPUTER SYSTEM (TCS) SUBSYSTEM



Source: A-Specs, April 1979

which a message is formatted and displayed. Flow charts showing the operations of the TCS in performing the four tasks in terms of these activities are presented in exhibits 2-16, 2-17, 2-18, and 2-19.

2.2.4 TCT

The TCT subsystem includes all TCT nodes in the network. The hardware structure of a TCT node is shown in exhibit 2-20. The component for which the performance measure is computed is shown in a shaded box. The TCT performs two tasks: (1) originating; and (2) terminating. There are four TCT activities: (1) inputting; (2) sending; (3) receiving; and (4) outputting. These tasks and activities are subsets of those of the TCS. The flow of operation to perform the two tasks are identical to the flow of operation of those tasks when performed by the TCS.

EXHIBIT 2-16: TCS FLOW OF OPERATION
WHEN ORIGINATING

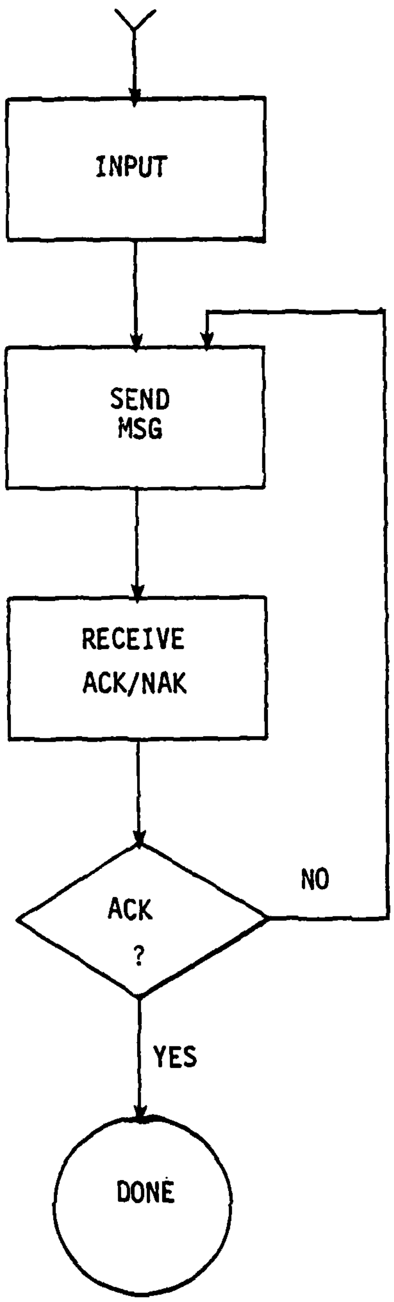


EXHIBIT 2-17: TCS FLOW OF OPERATION
WHEN TERMINATING

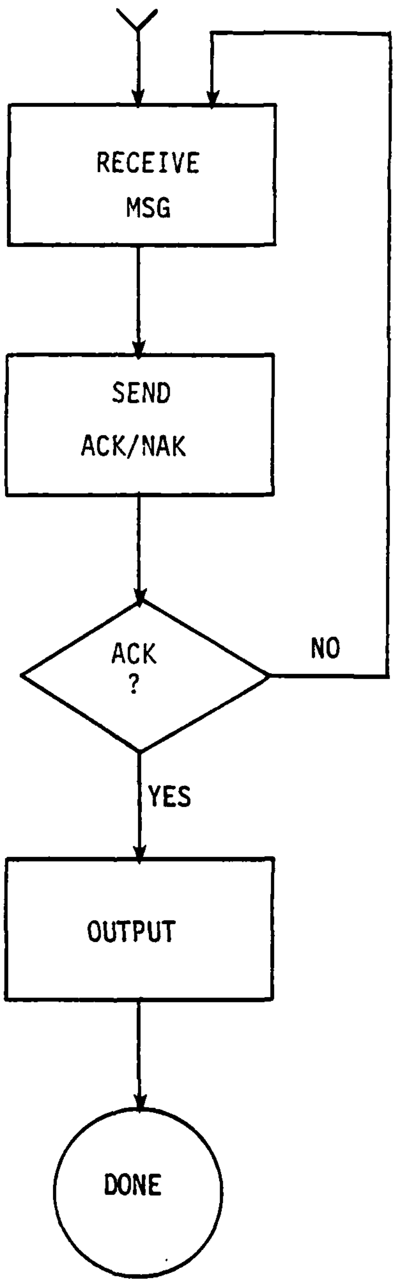


EXHIBIT 2-18: TCS FLOW OF OPERATION WHEN TRANSFERRING A MESSAGE TO THE DCC

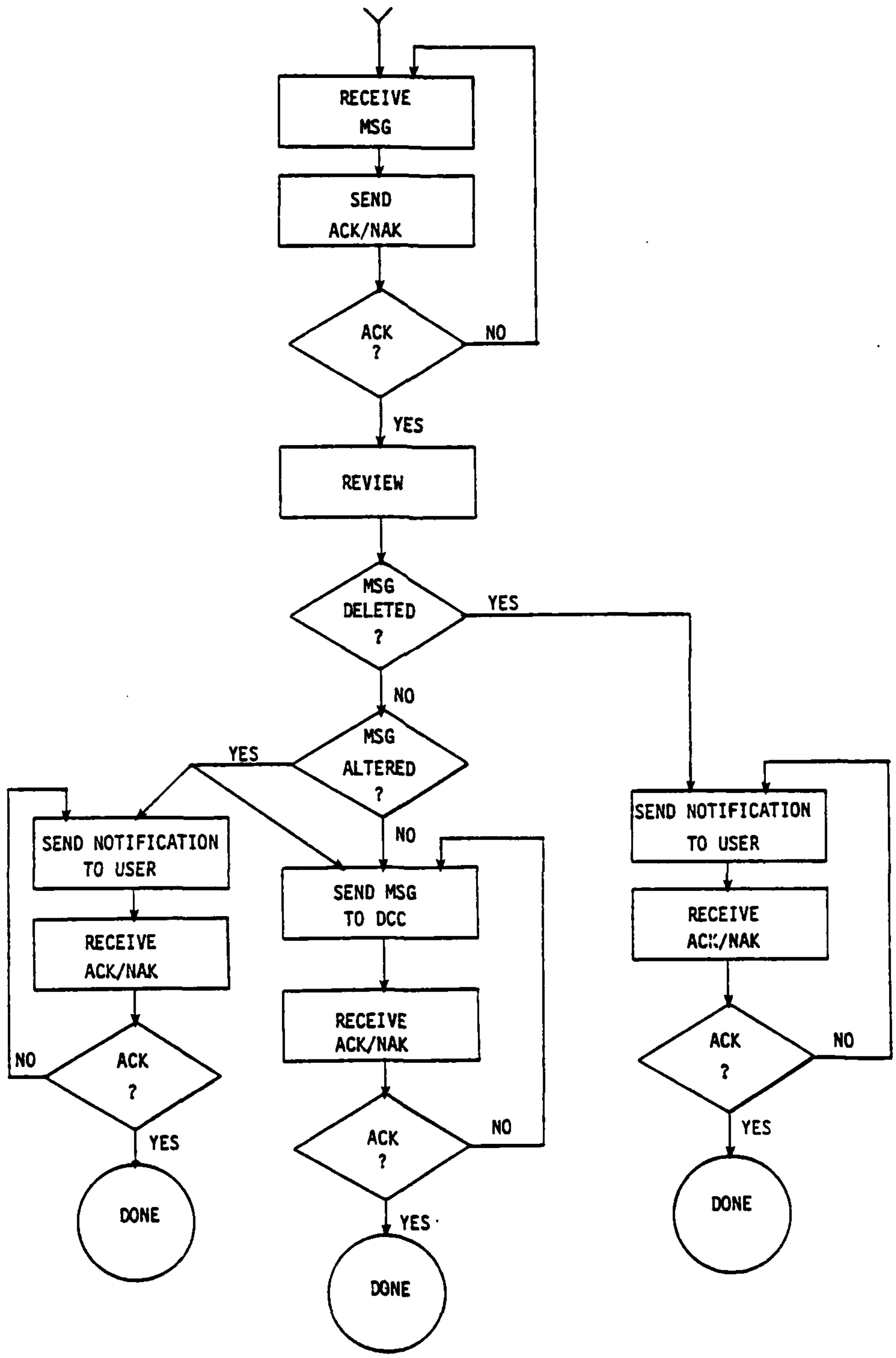


EXHIBIT 2-19: TCS FLOW OF OPERATION WHEN TRANSFERRING MESSAGE FROM DCC

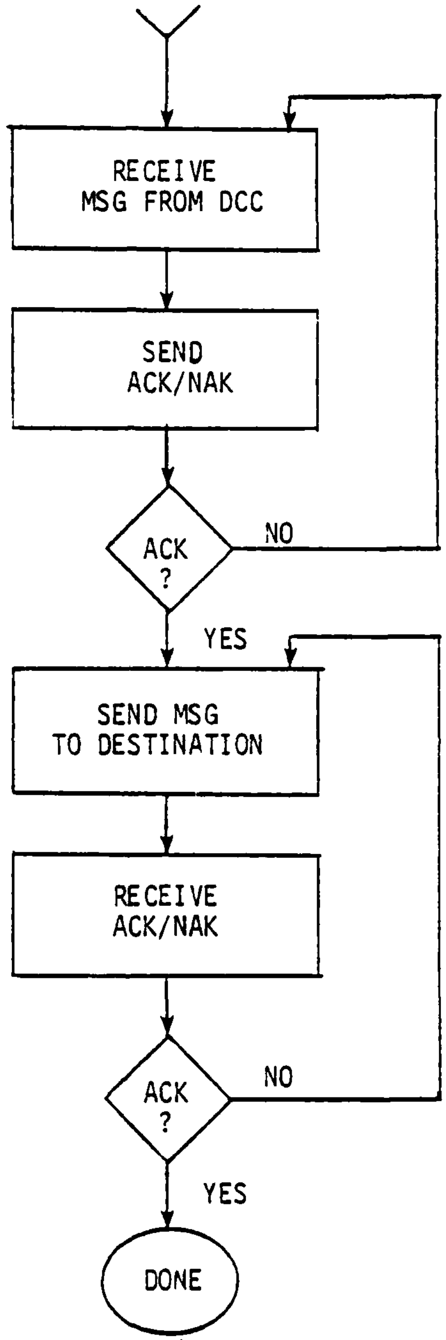
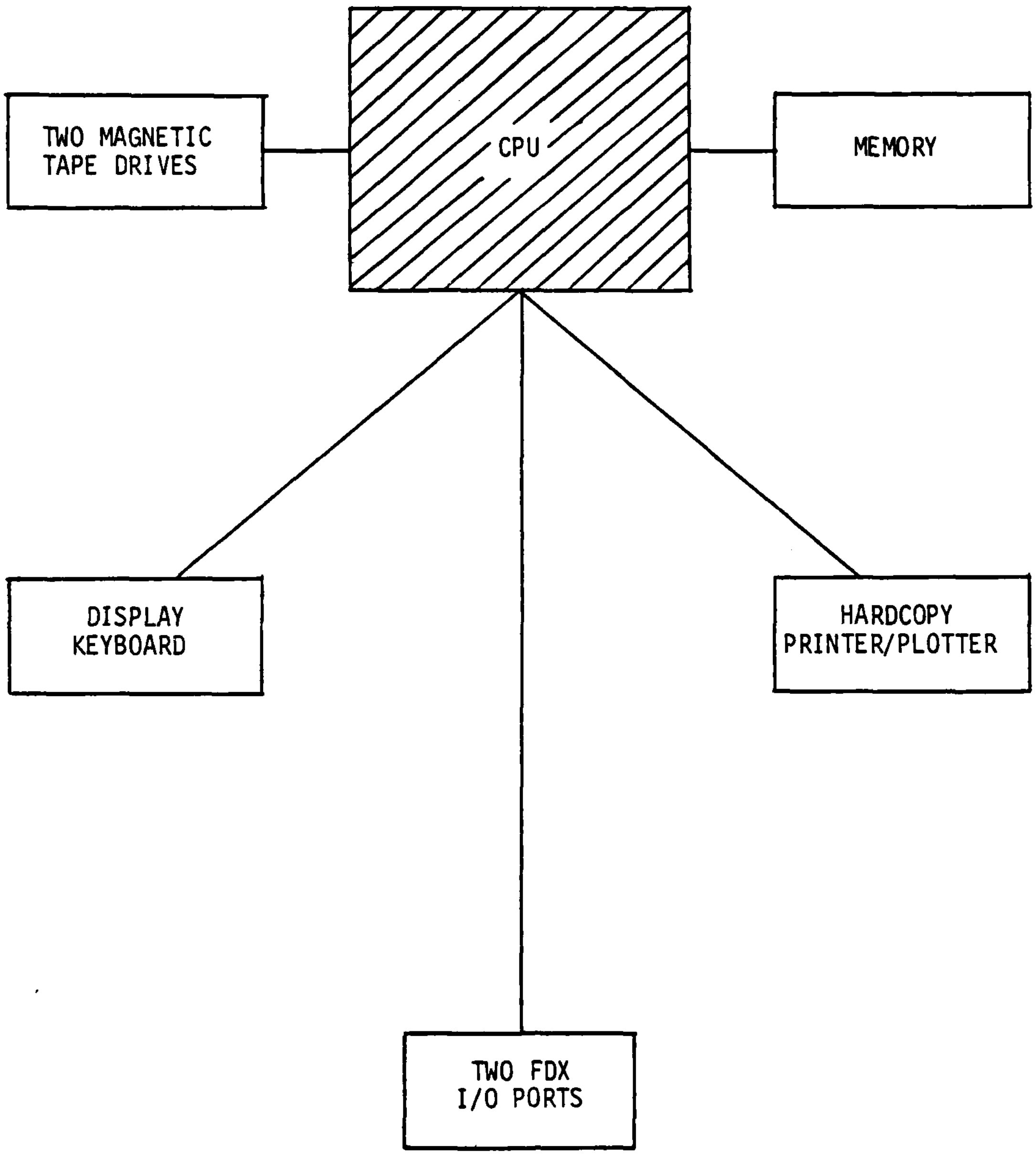


EXHIBIT 2-20: TACTICAL COMPUTER TERMINAL (TCT) SUBSYSTEM



Source: A-Specs, April 1979

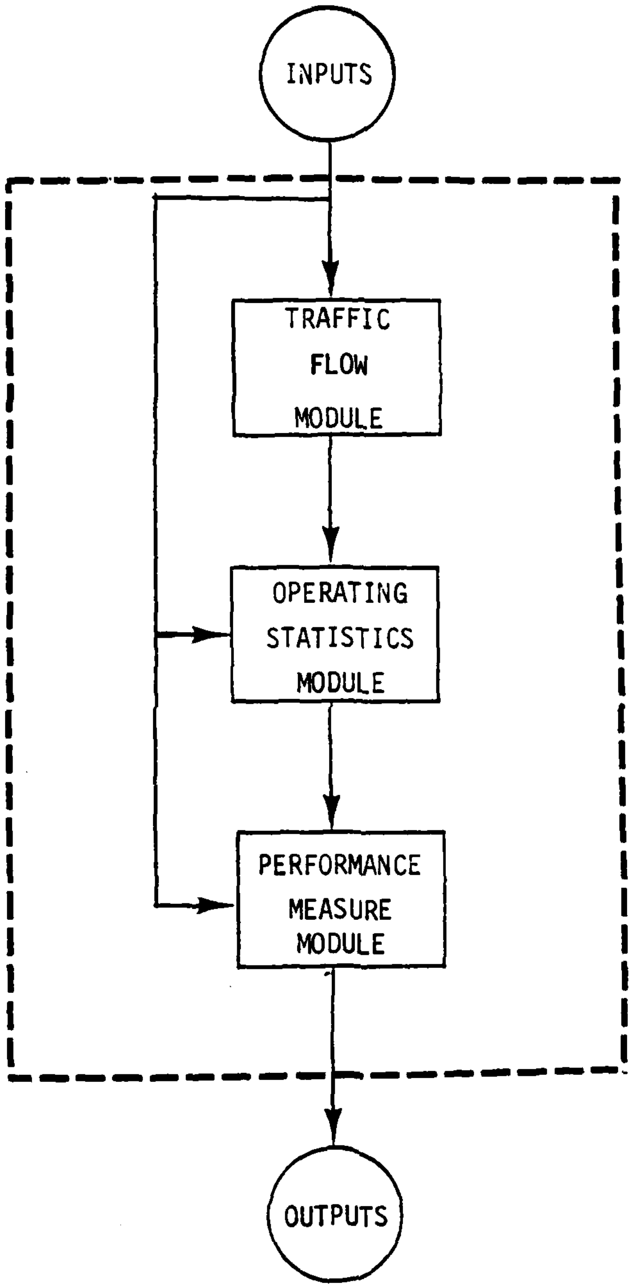
3.0 EQUATIONS AND DATA REQUIREMENTS

The representation of TOS described in the previous chapter was used to develop the mathematical model of TOS. The model is organized into three computational modules: (1) traffic flow; (2) operating statistics; and (3) performance measures. This organization is shown schematically in exhibit 3-1. For each type of message in each direction on each link in the network, the traffic flow module computes: (1) the arrival rate by type of message; and (2) the expected number of transmissions required for successful communication. The outputs of the operating statistics module are: (1) the aggregate arrival rate of messages of all types to each major component; (2) the mean service time per message at each major component; and (3) the second moment of the distribution of the service time per message at each major component. The outputs of the performance measure module are, for each major component: (1) its utilization; (2) the expected number of messages queued at the component; and (3) the expected length of time messages are delayed. The inputs, equations, and outputs of the three modules are discussed in the remainder of the chapter.

3.1 TRAFFIC FLOW MODULE

The traffic flow module consists of seven equations to compute the arrival rates of types of messages to each link in each direction, two equations to compute the expected number of transmissions for each type of message in each direction on each link, and two equations to compute the second moment of the distribution of the number of transmissions for each type of message in each direction on each link. The seven arrival

EXHIBIT 3-1: MODEL ORGANIZATION



rate equations are presented and briefly discussed in exhibit 3-2. The two equations for the expected number of transmissions to success are presented and discussed in exhibit 3-3. The two equations for the second moment of one number of transmissions are presented and discussed in exhibit 3-4. These latter equations are approximations since they take into consideration only the first moment of the distributions of the number of characters in each type of message, and not the higher moments. Since information on the higher moments would not be available to be used in the analysis for which the model was intended, the model was developed with those equations which did not require unavailable information.

3.2 OPERATING STATISTICS MODULE

The operating statistics module has three equations computing: (1) the aggregate arrival rate; (2) the mean service time; and (3) the second moment of the service time for each type of component for which the performance measures are computed. There are six types of components: (1) communication nets; (2) DBP CPU; (3) disk controller; (4) FEP CPU; (5) TCS CPU; and (6) TCT CPU. These equations are presented in exhibits 3-5 through 3-10.

3.3 PERFORMANCE MEASURE MODULE

The performance measure module consists of three equations to compute the utilization, the expected delay, and the expected queue length at each component. These equations are presented in exhibit 3-11.

EXHIBIT 3-2: ARRIVAL RATE EQUATIONS

OBJECTIVE: Compute L_{ijk} = the rate at which messages of type k arrive at the interface between node-i and the link connecting node-i with node-j.

WHERE:

i, j, s	are subscripts designating nodes;
x	is a subscript designating the DCC;
k	is a subscript designating a message type;
B_i	is the set of indirect nodes connected to the DCC through direct node-i; (e.g., a battalion TCT able to communicate with the DCC only through its parent brigade);
D	is the set of nodes directly connected to the DCC (e.g., a brigade with a direct link to the DCC);
R_{ik}	is the rate at which messages of type-k are originated by users at node-i. R_{ik} = 0 if k is not a user generated message or if i is the DCC;
P_{1ijk}	is the probability that a user-generated message of type-k sent from indirect node-i to direct node-j is deleted under hierarchical review;
P_{2ijk}	is the probability that a user generated message of type-k sent from indirect node-i to direct node-j is altered under hierarchical review;
$\bar{k}$	is the user-generated message type which produces output of type-k to the originator who sent the message of type-$\bar{k}$ (e.g., file queries);
$\bar{\bar{k}}$	is the user-generated message type which produces outputs of type-k to users other than the originator who sent the message of type-$\bar{\bar{k}}$ (e.g., file updates);

--Continued--

EXHIBIT 3-2: ARRIVAL RATE EQUATIONS

(Continued)

E_{1jk} is the expected number of outputs of type-k sent to node-j from the DCC as a result of an arrival at the DCC of message of type-$\overline{k}$ from node-j; and

E_{2jk} is the expected number of outputs of type-k sent to node-j from the DCC as a result of an arrival at the DCC of a message of type-$\overline{\overline{k}}$ from a node other than node-j.

CASE 1: i is a battalion (indirect);

j is a brigade (direct); and

k is a user-originated message.

$L_{ijk} = R_{ik}$

CASE 2: i is a brigade;

j is a battalion; and

k is a file update being returned to its originator following alteration or deletion under hierarchical review.

$L_{ijk} = L_{jik} \bullet (P_{1jik} + P_{2jik})$

CASE 3: i is a direct node with no indirect users;

x is the DCC; and

k is a user-originated message.

$L_{ixk} = R_{ik}$

-- Continued --

EXHIBIT 3-2: ARRIVAL RATE EQUATIONS

(Concluded)

CASE 4: i is a brigade;

x is the DCC; and

k is a user-originated message.

$$L_{ixk} = R_{ik} + \sum_{s \varepsilon B_i} R_{sk} (1 - P_{1sik})$$

CASE 5: i is a brigade;

j is a battalion (i.e., indirect); and

k is a DCC-generated message.

$$L_{ijk} = E_{1jk} \bullet (1 - P_{1jik}) \bullet R_{j\bar{k}} + E_{2jk} \left(\sum_{s \varepsilon B_i} L_{sx\bar{\bar{k}}} - R_{j\bar{\bar{k}}}(1 - P_{1ji\bar{\bar{k}}}) \right)$$

CASE 6: x is the DCC;

j is a direct user; and

k is a DCC-generated message.

$$L_{xjk} = E_{1jk} \bullet R_{j\bar{k}} + E_{2jk} \bullet \left(\sum_{s \varepsilon D_i} L_{sx\bar{\bar{k}}} - R_{i\bar{\bar{k}}} \right)$$

EXHIBIT 3-3: EQUATIONS FOR THE EXPECTED NUMBER OF TRANSMISSIONS TO SUCCESS

OBJECTIVE: Compute T_{ijk} = the expected number of transmissions to successful communication for a message of type-k from node-i to node-j

WHERE:

i,j are subscripts designating nodes

U_{ij} is the resulting bit error rate over the link from i to j after majority voting and before using the Hamming Code or other subsequent EDC procedures

B is the number of copies of the message sent at one time. If single blocking is used, B=1. If double blocking is used, B=2.

N_k is the mean number of characters in a message of type-k.

CASE 1: Hamming Code and multiple blocking EDC

$$T_{ijk} = 1/\left[1-\left(1-\left((1-U_{ij})^{12} + 12\cdot U_{ij}\cdot(1-U_{ij})^{11}\right)\right)^{B}\right]^{N_k}$$

CASE 2: Hamming Code, multiple blocking, and retained message copy EDC

$$T_{ijk} = \sum_{n=1}^{\infty} n\cdot\left[\left(1-\left(1-\left((1-U_{ijk})^{12} + 12\cdot U_{ij}\cdot(1-U_{ij})^{11}\right)\right)^{n\cdot B}\right)^{N_k} - \left(1-\left(1-\left((1-U_{ij})^{12} + 12\cdot U_{ij}\cdot(1-U_{ij})^{11}\right)\right)^{(n-1)\cdot B}\right)^{N_k}\right]$$

-- Continued --

EXHIBIT 3-3: EQUATIONS FOR THE EXPECTED NUMBER OF TRANSMISSIONS TO SUCCESS

(Continued)

DISCUSSION:

- U_{ij} is the total error rate over the link. If the link is on an FM net, a series of retransmission stations may be used. If there are n retransmission stations, there will be n+1 sublinks across the link. Empirical studies indicate that the total error rate over the link is roughly equal to the sum of the error rates over the sublinks.

Suppose: (1) there are n+1 sublinks; (2) the Sth sublink has error rate e_s when considered in isolation; and (3) the effects on a bit stream of transmission across a sublink are independent of the extent of prior transmission of the bit stream (e.g., there is no cumulative degradation of waveforms). Then the total error rate can be expressed in terms of the error rates on the sublinks:

- $$U_{ij} = 1 - \prod_{s=1}^{n+1} (1 - e_s) \cdot \left[1 + \sum_{r=2}^{\left[\frac{n+1}{2}\right]} \sum_{\substack{1 \le x_1 < \ldots \\ < x_{2r} \le n+1}} \prod_{y=1}^{2r} \left(e_{x_y} / (1 - e_{x_y}) \right) \right]$$

A first order approxmiation to the error rate across the link is:

- $$U_{ij} \approx 1 - \prod_{s=1}^{n+1} (1 - e_s).$$

These equations predict that when error rates are low, the error rate across a series of sublinks will be roughly equal to the sum of the error rates over the sublinks.

-- Continued --

EXHIBIT 3-3: EQUATIONS FOR THE EXPECTED NUMBER OF TRANSMISSIONS TO SUCCESS

(Concluded)

- $(1 - U_{ij})^{12} + 12 \cdot U_{ij} \cdot (1 - U_{ij})^{11}$

  is the probability that a character has 0 or 1 bits in error and is equal to the probability that the character is correct after Hamming Code EDC has been applied.

- $1 - \left(1 - \left((1 - U_{ij})^{12} + 12 \cdot U_{ij} \cdot (1 - U_{ij})^{11}\right)\right)^{B}$

  is the probability that a character is correct after Hamming Code and multiple blocking EDC have been applied. (Note: as stated earlier, the approximation is made that Hamming Code will detect any situation where there are errors in a character.)

- $\left(1 - \left(1 - \left((1 - U_{ij})^{12} + 12 \cdot U_{ij} \cdot (1 - U_{ij})^{11}\right)\right)^{B}\right)^{N_k}$

  is the probability that an entire message of N_k characters is correct after application of Hamming Code and multiple blocking.

EXHIBIT 3-4: EQUATIONS FOR THE SECOND MOMENT OF NUMBER OF TRANSMISSIONS TO SUCCESS

OBJECTIVE: Compute V_{ijk} = the second moment of the number of transmissions to successful communication for a message of type-k from node-i to node-j.

WHERE:

- i, j are subscripts designating nodes;
- U_{ij} is the resulting bit error rate over the link from i to j after majority voting and before using the Hamming Code or other subsequent EDC procedures;
- B is the number of copies of the message sent at one time. If single blocking is used, B=1. If double blocking is used, B=2;
- N_k is the mean number of characters in a message of type-k;
- T_{ijk} is the expected number of transmissions to success.

Case 1: Hamming Code and multiple blocking EDC

$$V_{ijk} = T_{ijk}^2 + T_{ijk}^2 (1 - 1/T_{ijk})$$

Case 2: Hamming Code, multiple blocking, and retained message copy EDC

$$V_{ijk} = \sum_{n=1}^{\infty} n^2 \cdot \left[\left(1 - \left(1 - \left((1 - u_{ij})^{12} + 12 \cdot u_{ij} \cdot (1 - u_{ij})^{11}\right)\right)^{n \cdot B}\right)^{N_k} - \left(1 - \left(1 - \left((1 - u_{ij})^{12} + 12 \cdot u_{ij} \cdot (1 - u_{ij})^{11}\right)\right)^{(n-1) \cdot B}\right)^{N_k} \right]$$

EXHIBIT 3-5: COMMUNICATIONS CHANNEL EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate and the first and second moments of the service time distribution for each communications channel. Note: full duplex channels are represented as two mono-directional channels.

WHERE:

A_r is the aggregate arrival rate of messages to channel-r;

M_{1r} is the mean service time of a message on channel-r;

M_{2r} is the second moment of the service time of a message on channel-r;

L_{ijk} is the arrival rate of messages of type-k from node-i to the interface between node-i and the link connecting node-i with node-j

T_{ijk} is the expected number of transmissions to success of a message of type-k across the link from i to j;

V_{ijk} is the second moment of the number of transmissions to success of a message of type-k across the link from i to j;

S_r is the set of all ordered pairs of nodes (i, j) that use communication channel-r as their link;

D_r is the keying sequence duration for channel-r;

R_r is the transmission rate for channel-r;

N_k is the mean number of characters in a message of type-k;

B is the number of copies of a message being sent at one time; and

$\underline{K}$ is the set of all message types.

-- Continued --

EXHIBIT 3-5: COMMUNICATIONS CHANNEL EQUATIONS
(Concluded)

EQUATION 1:

- $A_r = \sum_{(i,j)\varepsilon S_r} L_{ijk}$

EQUATION 2:

- $$M_{1r} = \sum_{(i,j)\varepsilon S_r} \sum_{k\varepsilon \mathbb{K}} \Big(L_{ijk} \cdot T_{ijk} \cdot (D_r + N_k/R_r) + L_{jik} \cdot T_{jik} \cdot (D_r + 16/R_r) \Big)/A_r$$

EQUATION 3:

- $$M_{zr} = \sum_{(i,j)\varepsilon S_r} \sum_{k\varepsilon \mathbb{K}} \Big(L_{ijk} \cdot V_{ijk} \cdot (D_r + N_k/R_r)^2 + L_{jik} \cdot V_{jik} \cdot (D_r + 16/R_r)^2 \Big)/A_r$$

DISCUSSION:

- Link control/response messages are 16 characters in length. $(D_r + 16/R_r)$ is the time to transmit one ACK or NAK.

EXHIBIT 3-6: DBP EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate (including DCC-originated messages as arrivals), mean service time, and second moment of the service time of messages at the DBP CPU.

WHERE:

- A is the aggregate arrival rate;
- M_1 is the mean service time;
- M_2 is the second moment of the service time;
- x is a subscript denoting the DCC;
- L_{ijk} is the arrival rate of messages of type-k from node-i to the interface between node-i and the link connecting node-i with node-j;
- S_{1k} is the mean service time at the DBP CPU of messages of type-k;
- S_{2k} is the second moment of the service time of messages of type-k at the DBP CPU;
- D is the set of all nodes directly connected to the DCC, and
- K is the set of all message types.

EQUATION 1:

$$A = \sum_{i \epsilon D} \sum_{k \epsilon K} (L_{ixk} + L_{xik})$$

EQUATION 2:

$$M_1 = \sum_{i \epsilon D} \sum_{k \epsilon K} S_{1k} (L_{ixk} + L_{xik}) / A$$

-- Continued --

EXHIBIT 3-6: DBP EQUATIONS

(Concluded)

EQUATION 2:

$$M_2 = \sum_{i\varepsilon D} \sum_{k\varepsilon \underline{K}} S_{2k} (L_{ixk} + L_{xik}) /A$$

DISCUSSION:

- For all k, $S_{2k} \geq S_{1k}^{2}$.

EXHIBIT 3-7: FEP EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate of messages to the FEP, the mean service time of a message at the FEP, and the second moment of the service time of a message at the FEP.

WHERE:

- A is the aggregate arrival rate;
- M_1 is the mean service time;
- M_2 is the second moment of the service time;
- x is a subscript denoting the DCC;
- D is the set of all nodes directly connected to the DCC;
- $\mathbb{K}$ is the set of all message types;
- L_{ijk} is the rate at which messages of type-k from node-i arrive to the interface between node-i and the link connecting node-i with node-j;
- T_{ijk} is the expected number of transmissions to success of a message of type-k over the link from i to j;
- V_{ijk} is the second moment of the number of transmissions to success of a message of type-k over the link from i to j;
- S_{1k} is mean FEP CPU time to process an incoming message of type-k;
- S_{2k} is the mean FEP CPU time spent every time a copy of a message of type-k is transmitted or retransmitted to the FEP;
- S_{3k} is the mean FEP CPU time spent every time a copy of a message of type-k is transmitted or retransmitted from the FEP to a node;
- S_{4k} is the mean FEP CPU overhead time spent on a DCC generated message of type-k.

-- Continued --

EXHIBIT 3-7: FEP EQUATIONS

(Concluded)

EQUATION 1:

$$A = \sum_{i \in D} \sum_{k \in K} (L_{ixk} + L_{xik})$$

EQUATION 2:

$$M_1 = \sum_{i \in D} \sum_{k \in K} \left(L_{ixk} \cdot (S_{1k} + T_{ixk} \cdot S_{2k}) + L_{xik} \cdot (S_{4k} + T_{xik} \cdot S_{3k})\right) / A$$

EQUATION 3:

$$M_2 = \sum_{i \in D} \sum_{k \in K} \left(L_{ixk} (S_{1k} + T_{ixk} \cdot S_{2k})^2 + L_{xik} (S_{4k} + T_{xik} \cdot S_{3k})^2\right) / A$$

DISCUSSION:

Equation 3 uses $(S_{1k} + T_{ixk} \cdot S_{2k})^2$ as an approximation of the second moment of the FEP service time for a message of type-k from node-i since data on the second moment of the FEP service time was not likely to be available.

EXHIBIT 3-8: DISK CONTROLLER EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate of messages, mean service time, and second moment of the service time at the message disk and data disk.

WHERE:

A_i is the aggregate message arrival rate at i, i=1 denotes the message disk, i=2 denotes the data disk;

M_{1i} is the mean service time for a message at the disk;

M_{2i} is the second moment of the service time for a message at the disk;

N_{ik} is the mean number of times a read or write on disk-i occurs as a result of a message of type-k at the DCC;

S is the mean disk access time (mean rotational latency plus mean positional latency);

x is a subscript denoting the DCC;

D is the set of all nodes directly connected to the DCC; and

$\mathbb{K}$ is the set of all message types.

EQUATION 1:

$$A_i = \sum_{j \in D} \sum_{k \in \mathbb{K}} (L_{jxk} + L_{xjk}) \qquad \text{for } i=1,2$$

-- Continued --

EXHIBIT 3-8: DISK CONTROLLER EQUATIONS

(Concluded)

EQUATION 2:

$$M_{1i} = \sum_{j \epsilon D} \sum_{k \epsilon K} S \cdot N_{ik} \cdot (L_{jxk} + L_{xjk}) / A_i$$

EQUATION 3:

$$M_{2i} = \sum_{j \epsilon D} \sum_{k \epsilon K} S^2 \cdot N_{ik}^2 \cdot (L_{jxk} + L_{xjk}) / A_i$$

DISCUSSION:

- The mean access time is used as an approximation of the mean access plus read/write time since, for movable head disks such as used by TOS, the access time is very large when compared to the read/write time for messages several hundred characters in length. For the TOS disks the manufacturer's estimate of the mean access time is 38.3 ms.

- $S^2 \cdot N_{ik}^2$ is used as an estimate of the second moment of the disk service time due to lack of data.

EXHIBIT 3-9: TCS EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate of messages, mean service time per message, and service time per message at each TCS.

WHERE:

A_i is the aggregate message arrival rate at node-i;

M_{1i} is the mean service time at the TCS of node-i (when i has a TCS).

M_{2i} is the second moment of the service time at the TCS of node-i i (when node-i has a TCS).

L_{ijk} is the rate at which messages of type k from node i arrive to the interface between node-i and the link connecting node-i with node-j

T_{ijk} is the expected number of transmissions to success for a message of type-k sent from i to j

V_{ijk} is the second moment of the number of transmissions to success for a message of type-k from i to j.

S_{1k} is the mean TCS CPU time occupied whenever a message of type-k is input

S_{2k} is the mean TCS CPU time spent every time a message of type-k sent from the TCS is transmitted or retransmitted from the TCS

S_{3k} is the mean TCS CPU time spent for review of a message of type k from an indirect node being transferred to the DCC

S_{4k} is the mean TCS CPU time occupied whenever a message of type k is terminated at the node.

S_{5k} is the mean TCS CPU time spent every time a message being received is transmitted or retransmitted to the TCS

-- Continued --

EXHIBIT 3-9: TCS EQUATIONS

(Continued)

x is a subscript denoting the DCC

B_i is the set of all indirect nodes connected to the DCC through node-i

K is the set of all message types

R_{ik} is the rate at which message of type-k are entered at node-i

P_{1ijk} is the probability that a message of type-k, from node-i through node-j is deleted under hierarchial review

P_{2ijk} is the probability that a message of type-k from node-i through node-j is altered under hierarchical review

---

EQUATION 1:

$$A_i = \sum_{k \varepsilon K} (R_{ik} + L_{xik} + \sum_{r \varepsilon B_i} L_{rik})$$

EQUATION 2:

$$M_{1i} = \sum_{k \varepsilon K} \Big[R_{ik} \cdot (S_{1k} + T_{ixk} \cdot S_{2k}) +$$

$$\sum_{r \varepsilon B_i} L_{rik} \cdot P_{2rik} \cdot (S_{5k} \cdot T_{rik} + S_{1k} + S_{3k} + S_{4k} + S_{2k} \cdot T_{irk} + S_{2k} \cdot T_{ixk}) +$$

$$\sum_{r \varepsilon B_i} L_{rik} \cdot P_{1rik} \cdot (S_{5k} \cdot T_{rik} + S_{4k} + S_{1k} + S_{2k} T_{irk} + S_{3k}) +$$

$$\sum_{r \varepsilon B_i} L_{rik} \cdot (1 - P_{1rik} - P_{2rik}) \cdot (S_{5k} \cdot T_{rik} + S_{3k} + S_{2k} \cdot T_{ixk}) +$$

$$\sum_{r \varepsilon B_i} ((L_{irk} - L_{rik} \cdot (P_{1rik} + P_{2rik})) \cdot (S_{2k} \cdot T_{irk} + S_{5k} \cdot T_{xik})) +$$

$$(L_{xik} - \sum_{r \varepsilon B_i} (L_{irk} - L_{rik} \cdot (P_{1rik} + P_{2rik}))) \cdot (S_{4k} + S_{5k} \cdot T_{xik}) \Big] / A_i$$

-- Continued --

EXHIBIT 3-9: TCS EQUATIONS

(Concluded)

EQUATION 3:

$$
\begin{aligned}
M_{2i} = \sum_{k \varepsilon K} \Big[& R_{ik} \cdot (S_{1k} + T_{ixk} \cdot S_{2k})^2 + \\
& \sum_{r \varepsilon B_i} L_{rik} \cdot P_{2rik} \cdot (S_{5k} \cdot T_{rik} + S_{1k} + S_{3k} + S_{4k} + S_{2k} \cdot T_{irk} + S_{2k} \cdot T_{ixk})^2 + \\
& \sum_{r \varepsilon B_i} L_{rik} \cdot P_{1rik} \cdot (S_{5k} \cdot T_{rik} + S_{4k} + S_{1k} + S_{2k} \cdot T_{irk} + S_{3k})^2 + \\
& \sum_{r \varepsilon B_i} L_{rik} \cdot (1 - P_{1rik} - P_{2rik}) \cdot (S_{5k} \cdot T_{rik} + S_{3k} + S_{2k} \cdot T_{ixk})^2 + \\
& \sum_{r \varepsilon B_i} ((L_{irk} - L_{rik} \cdot (P_{1rik} + P_{2rik}) \cdot (S_{2k} \cdot T_{irk} + S_{5k} \cdot T_{xik}))^2 + \\
& (L_{xik} - \sum_{r \varepsilon B_i} (L_{irk} - L_{rik} (P_{1rik} + P_{2rik}))) \cdot (S_{4k} + S_{5k} \cdot T_{xik})^2 \Big] / A_i
\end{aligned}
$$

DISCUSSION:

- For each type of processing on each message type, Eq. 3 uses the mean service time squared as an approximation to the second moment of the service time. This approximation is a lower bound and is made in lieu of data on the second moment of the service times.

EXHIBIT 3-10: TCT EQUATIONS

OBJECTIVE: Compute the aggregate arrival rate of messages, mean service time per message, and service time per message at each TCT.

WHERE:

A_i is the aggregate arrival rate to node-i;

M_{1i} is the mean service time at TCT of node-i (when i has a TCT);

M_{2i} is the second moment of the service time at the TCT of node-i (when node-i has a TCT);

L_{ijk} is the rate at which messages of type-k from node-i arrive to the interface between node-i and the link connecting node-i with node-j;

T_{ijk} is the expected number of transmissions to success for a message of type-k sent from i to j;

V_{ijk} is the second moment of the number of transmissions to success for a message of type-k from i to j;

S_{1k} is the mean TCT CPU time occupied whenever a message of type-k is input;

S_{2k} is the mean TCT CPU time spent every time a message of type-k sent from the TCS is transmitted or retransmitted from the TCS;

S_{3k} is the mean TCT CPU time occupied whenever a message of type-k is terminated at the node;

S_{4k} is the mean TCT CPU time spent every time a message being received is transmitted or retransmitted to the TCT;

$\mathbb{K}$ is the set of all message types;

R_{ik} is the rate that messages of type-k are entered at node-i;

y is a subscript denoting the node to which the TCT is connected (either the DCC or a brigade TCS);

-- Continued --

EXHIBIT 3-10: TCT EQUATIONS

(Concluded)

EQUATION 1:

$$A_i = \sum_{k \in K} (L_{yik} + R_{ik})$$

EQUATION 2:

$$M_{1i} = \sum_{k \in K} \left(R_{ik} \cdot (S_{1k} + T_{iyk} \cdot S_{2k}) + L_{yik} \cdot (S_{3k} + T_{yik} \cdot S_{4k})\right) / A_i$$

EQUATION 3:

$$M_{2i} = \sum_{k \in K} \left(R_{ik} \cdot (S_{1k} + T_{iyk} \cdot S_{2k})^2 + L_{yik} \cdot (S_{3k} + T_{yik} \cdot S_{4k})^2\right) / A_i$$

Discussion:

Eq. 3 is an approximation made in lieu of data availability on the second moments of TCT service times.

EXHIBIT 3-11: PERFORMANCE MEASURE EQUATIONS FOR COMPONENTS

EQUATION 1:

$\rho = \lambda / \mu$, where

ρ = utilization

λ = arrival rate

μ = service rate = 1/(mean service time)

EQUATION 2:

$d = 1/\mu + \lambda \cdot V/(2 \cdot (1-\rho))$, where

d = expected delay

V = second moment of the service time distribution

EQUATION 3:

$q = \lambda^2 . V/(2 \cdot (1-p))$ / where

q = the expected number of messages waiting for service

DISCUSSION:

- Equations 2 and 3 are meaningful only when the utilization of the component is less than 100 percent.

-- Continued --

The equation for utilization is valid for any arrival and service distributions; however, the equations for the two other performance measures computed by the model, expected delay, and expected queue length at a component are derived from an M/G/1 representation of queueing at the component. An M/G/1 queueing system has a Poisson arrival stream (M),[1] an arbitrary service time distribution, a single server and first in first out queueing discipline. In general, the departure stream from an M/G/1 queue is not Poisson. Consequently, in an arbitrary network of single server queues, the arrival streams to the queues will not be Poisson. Equations derived from an M/G/1 representation will provide good approximations for expected delay and expected queue length, provided one of the following conditions are met: (1) the server of the queue has a low utilization; or (2) the arrival stream resembles a Poisson arrival stream.

When the server of a queue has a low utilization, the time between arrivals is very large compared to the service time. For example, if the utilization of the server equals 0.1, then the mean time between arrivals will be ten times the mean service time. Consequently, the expected delay is close to the mean service time, and the expected queue length is close to zero. Furthermore, because very little actual queueing occurs, departure stream will be very similar to the arrival stream. If the arrival stream is Poisson, the departure stream will be approximately Poisson regardless of the service time distribution.

[1]In a Poisson arrival stream, the interarrival times have a negative exponential distribution.

The arrival stream to a node will resemble a Poisson arrival stream if the arrival stream consists of the pooled departure streams from a number of nodes.[1] By the above result, the pooled departure streams can be passed through a number of servers in series and still retain their resemblance to a Poisson process provided the servers in series all have low utilizations. This is true even when the individual processes that are being pooled are far from Poisson in nature provided that no one of the arrival streams being pooled has an arrival rate much greater than any of the others.

The significance of these results for the TOS model is that they strongly support the use of the M/G/1 queue approximation of queueing at the major components. When the model was exercised with the best available estimates of the input values, all components except the FM nets and multichannel frequencies had utilizations less than 0.1.[2] These components would, therefore, have expected delays and expected queue lengths very close to those predicted by the M/G/1 approximation, and would have output streams with distributions almost identical to their input streams. The FM nets and multichannel frequencies did have high utilizations at some error rates, but also had sufficient pooling to justify the M/G/1 approximation. The pooling on multichannel occurs because the DCC-generated outputs are responses to the pooled inputs of all users (30-40 users, depending on the configuration). The arrival stream to a brigade FM net will pool the input stream of each battalion, the output streams to each battalion, and hierarchical review notifications.

---

[1]See appendix A, Queueing Systems with Pooled Arrival Streams, for a detailed explanation.

[2]See ARI Research Notes 80-12 for details.

APPENDIX A

QUEUEING SYSTEMS WITH POOLED ARRIVAL STREAM

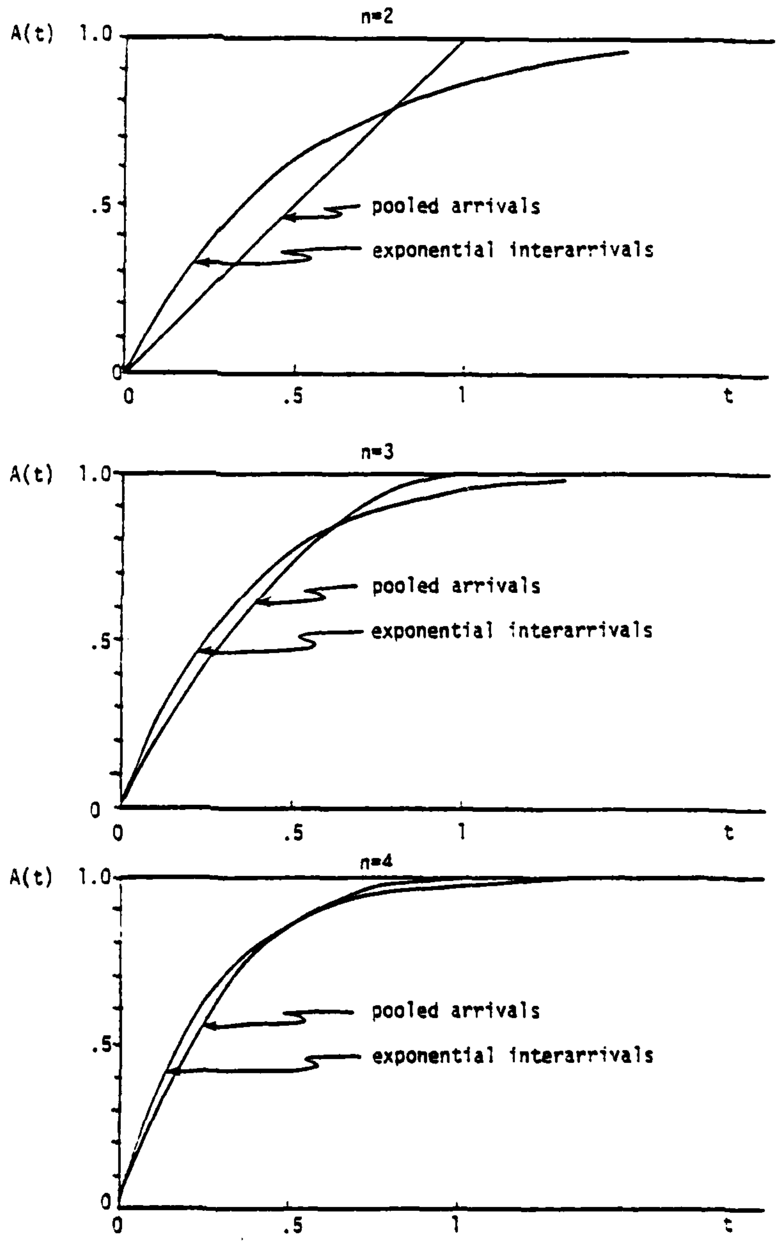
The pooling of arrival streams is a factor that contributes heavily to the approximate validity of a Poisson arrival stream to nodes in queueing networks. This is true even when the individual processes that are being pooled are far from Poisson in nature. If "n" identical and independent arrival processes are superimposed, the distribution of the time between arrivals for the pooled stream is given by:

$$a(t) = -\frac{d}{dt}\left\{\left[1-F(t)\right]\left[\int_t^{\infty}\frac{1-F(x)}{\bar{F}}\,dx\right]^{n-1}\right\}, \quad t \geq 0.$$

If the distribution of time between arrivals for each component process in the pooled stream is $f(t) = \lambda \cdot \exp(-\lambda t)$, then $a(t) = n\lambda \cdot \exp(-n\lambda t)$. Therefore, if "n" identical Poisson processes are pooled, the resulting process is also a Poisson process with arrival rate $n\lambda$. If "n" deterministic processes are pooled, each with time "c" between arrivals, then $a(t) = (n-1)/c \cdot (1-t/c)^{n-2}$ for $n \geq 2$ and $0 \leq t \leq c$. It is a well-known result of renewal theory that as $n \to \infty$, $a(t)$ approaches the negative exponential distribution. This explains why the arrival process to a node in a queueing network often resembles a Poisson process.

Exhibit A-1 compares the cumulative distribution functions of pooled deterministic arrival streams to exponential interarrivals for n = 2, 3, and 4. (For simplicity, "c" has been assigned the value one.) Note that

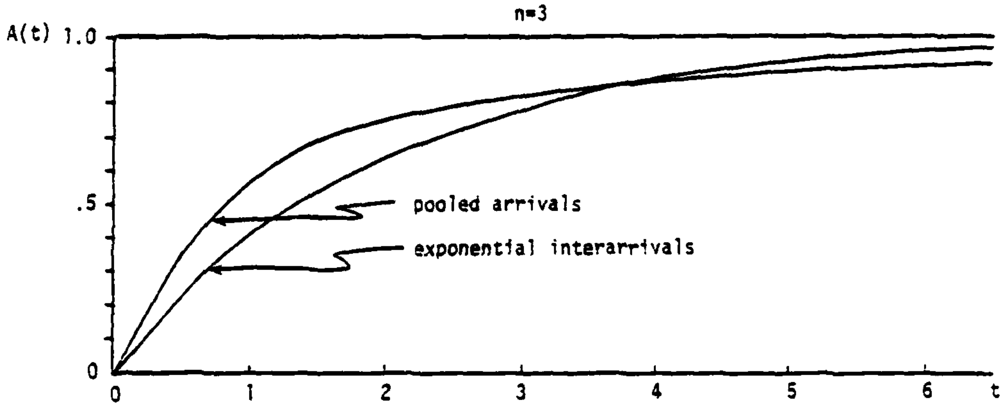
EXHIBIT A-1: COMPARISON OF POOLED DETERMINISTIC ARRIVALS WITH THE POISSON PROCESS



for n = 4, the pooled arrival stream already closely resembles a Poisson process. For distributions f(t) with a coefficient of variation between zero and one, by pooling as few as four identical arrival streams into one, a(t) becomes a close approximation to the exponential distribution. When the coefficient of variation is greater than one, we might expect that in order to obtain exponential behavior, the number of component processes would have to increase as the coefficient of variation increases. From simulation results, it appears that to obtain approximately exponential interarrivals, "n" must be on the order of the coefficient of variation of f(t), at least when f(t) corresponds to a mixture of shifted geometrics with a coefficient of variation greater than one. This compares favorably with the analytic results obtained for a hyperexponential distribution with coefficient of variation greater than one. The cumulative distributions for pooled hyperexponential arrival streams with n = 3 and 4 are compared with exponential interarrivals in exhibit A-2. (The coefficient of variation for this particular hyperexponential distribution is 3.2). For n = 4, the pooled process closely resembles the negative exponential.

Returning to the pooling of deterministic arrival streams, we examine the robustness of the M/M/1 queue by comparing the average system time in a G/M/1 queue whose arrival stream is obtained by pooling "n" deterministics, denoted Dn/M/1, with the average system time in an M/M/1 queue. In exhibit A-3, the ratio of average system time for the M/M/1 to the average system time for the Dn/M/1 queue is plotted against "n". At high utilization (0.95), the M/M/1 model predicts the mean system time to be twice the actual value for a deterministic arrival stream (n = 1). For n = 5, however, the M/M/1 model's prediction at high utilization is

EXHIBIT A-2: COMPARISON OF POOLED HYPEREXPONENTIAL ARRIVALS WITH THE POISSON PROCESS



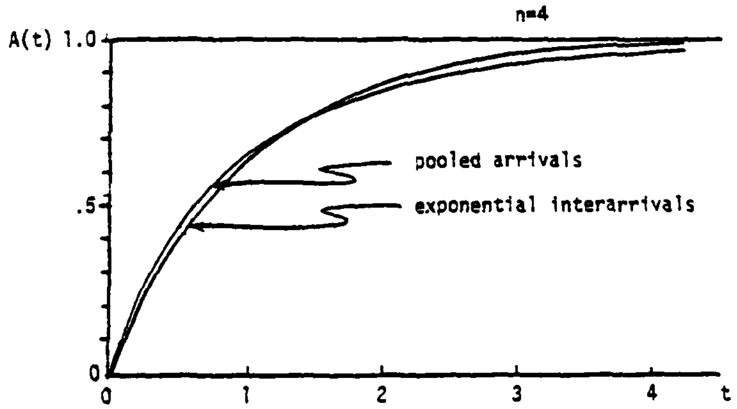
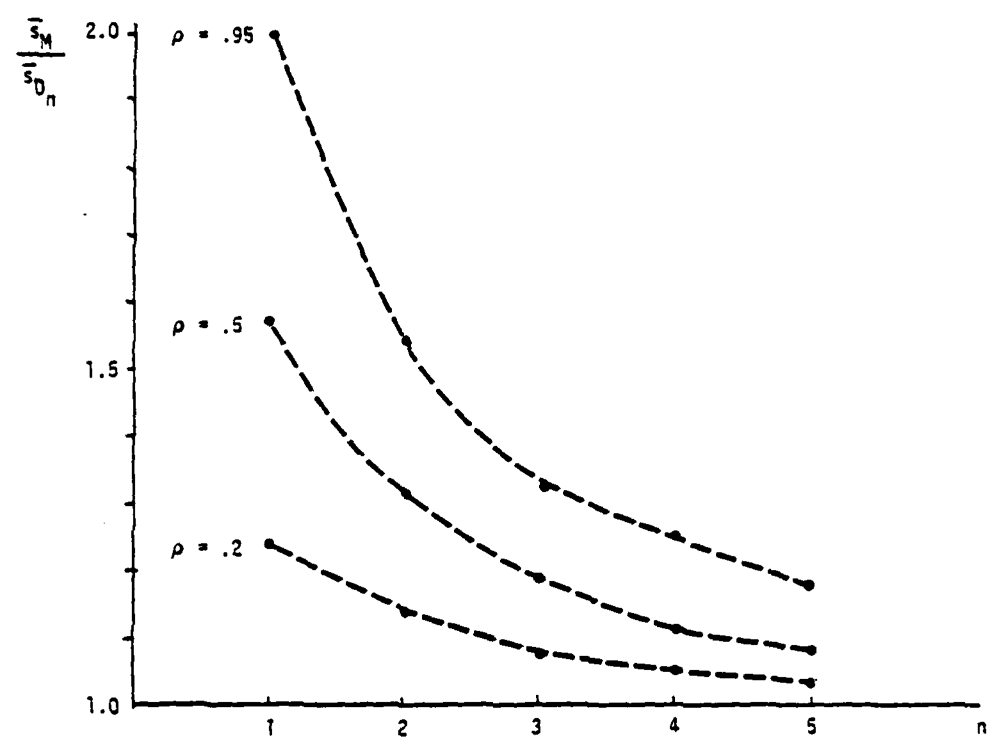


EXHIBIT A-3: RATIO OF AVERAGE SYSTEM TIME IN THE M/M/1 QUEUE TO AVERAGE SYSTEM TIME IN THE D_n/M/1 QUEUE VS. n



only 17 percent greater than the actual value. At lower utilization, system behavior approaches that of an M/M/1 queue more quickly.

The significance of these results for networks of queues is that, if sufficient pooling takes place, the arrival processes to each node in the network should look approximately Poisson, irrespective of the form of the departure process from each node that contributes to the merged stream. Each node in the network could then be analyzed as an independent M/G/1 queue. Simulation results for simplified versions of a TOS system show this approximation to be relatively good when sufficient pooling takes place.

APPENDIX B: GLOSSARY OF ACRONYMS

A & P	-	Analysis & Production
ACK	-	Acknowledgement
ADA BN	-	Air Defense Artillery Battalion
ADP	-	Automated Data Processing
ARM CAV	-	Armored Calvary
ARQ	-	Automatic Retransmission on Request
AVN BN	-	Aviation Battalion
BDE	-	Brigade
BIR	-	Battlefield Information Report
BPS	-	Bits Per Second
BN	-	Battalion
CAV	-	Cavalry
CCC	-	Computer Control Console
CDR	-	Commander
CEWI	-	Combat Electronic Warfare Intelligence
CIM	-	Communications Interface Module
CMS	-	Communications Management System
COMSEC	-	Communications Security
CONOPS	-	Continuity of Operations
CP	-	Command Post
CPU	-	Central Processing Unit
D/K/P	-	Display/Keyboard/Printer
D/L	-	Distribution List
DAME	-	Division Airspace Management Element
DB	-	Double Blocking

APPENDIX B: GLOSSARY OF ACRONYMS

(continued)

DBMS	-	Data Base Management System
DBP	-	Data Base Processor
DCC	-	Division Computer Center
DDA	-	Design/Decision Aid
DIOM	-	Data Input/Output Module
DISCOM	-	Division Support Command
DIV ARTY	-	Division Artillery
DTOC	-	Division Tactical Operations Center
DTOS	-	Division Tactical Operations System
E/V	-	Edit and Validation
EDC	-	Error Detection and Correction
EMI	-	Electromagnetic Interference
ENGR BN	-	Engineering Battalion
ENQ	-	Enquiry
ENSIT	-	Enemy Situation
EOB	-	Enemy Order of Battle
ESD	-	Enemy Situation Data
EW	-	Electronic Warfare
FDC	-	Fire Detection Center
FDX	-	Full Duplex
FEC	-	Forward Error Correction
FEP	-	Front End Processor
FM	-	Frequency Modulation
FMS	-	File Management System

APPENDIX B: GLOSSARY OF ACRONYMS

(continued)

FRENSIT	-	Friendly Situation
FSE	-	Fire Support Element
FSK	-	Frequency Shift Keying
HDX	-	Half Duplex
I/O	-	Input/Output
ICM	-	Intelligence Collection Management
IDS	-	Interactive Display System
IMR	-	Incoming Message Retrieval
INTEL	-	Intelligence
IOI	-	Input/Output Interface
IOU	-	Input/Output Unit
KBPS	-	Kilobits Per Second
MCMU	-	Mass Core Memory Unit
MLM	-	Memory Loading Module
MLU	-	Memory Load Unit
MSG	-	Message
NAI	-	Named Area of Interest
NAK	-	Non-Acknowledgement
NATO	-	North Atlantic Treaty Organization
OPS	-	Operations
OS	-	Operating System
PLM	-	Preloaded Message
PM	-	Project Manager
RMC	-	Retained Message Copy

APPENDIX B: GLOSSARY OF ACRONYMS

(concluded)

SB	-	Single Blocking
SDL	-	Standard Distribution Lists
SIU	-	System Interface Unit
SOP	-	Standard Operating Procedure
SQN	-	Squadron
SRI	-	Standing Request for Information
SWF	-	Staff Working File
SYSCON	-	System Controller
TAC CP	-	Tactical Command Post
TACFIRE	-	Tactical Fire Direction System
TB	-	Triple Blocking
TCS	-	Tactical Computer System
TCT	-	Tactical Computer Terminal
TCU	-	Terminal Control Unit
TDC	-	Time Dispersed Coding
TER	-	Terrain File
TOC	-	Tactical Operations Center
TOS	-	Tactical Operations System
UTD	-	Unit Tactical Disposition
UTO	-	Unit Task Organization
XMSN	-	Transmission